



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV MATEMATIKY

INSTITUTE OF MATHEMATICS

NEWTON A NUMERICKÁ MATEMATIKA

NEWTON AND NUMERICAL MATHEMATICS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Štěpán Obrátil

VEDOUCÍ PRÁCE

SUPERVISOR

Mgr. Jitka Zatočilová, Ph.D.

BRNO 2017

Zadání bakalářské práce

Ústav: Ústav matematiky
Student: **Štěpán Obrátil**
Studijní program: Aplikované vědy v inženýrství
Studijní obor: Matematické inženýrství
Vedoucí práce: **Mgr. Jitka Zatočilová, Ph.D.**
Akademický rok: 2016/17

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Newton a numerická matematika

Stručná charakteristika problematiky úkolu:

Sir Isaac Newton bývá často považován za jednu z nejvlivnějších osob v dějinách lidstva. V numerické matematice je jeho jméno spojeno např. s numerickým integrováním, s řešením soustav nelineárních rovnic, s optimalizačními metodami a mnohými dalšími.

Cíle bakalářské práce:

Uvedení Newtonových numerických metod a jejich modifikací, jejich aplikace na příkladech a srovnání s jinými adekvátními numerickými metodami.

Seznam doporučené literatury:

HOROVÁ, I. a J. ZELINKA. Numerické metody. Brno: Masarykova univerzita v Brně. 2004.

DĚMIDOVIČ, B. P. a I. A. MARON. Základy numerické matematiky. Praha : Státní nakladatelství technické literatury. 1966.

ČERNÁ, R., M. MACHLICKÝ, J. VOGEL a Č. ZLATNÍK. Základy numerické matematiky a programování. Praha : Státní nakladatelství technické literatury. 1987.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2016/17

V Brně, dne

L. S.

prof. RNDr. Josef Šlapal, CSc.
ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
děkan fakulty

Abstrakt

Tématem bakalářské práce jsou Newtonovy metody pro numerické řešení různých problémů. Zejména je vysvětlena problematika řešení nelineárních rovnic, soustav nelineárních rovnic a numerický výpočet integrálů. Je předvedena Newtonova metoda pro řešení nelineárních rovnic a mnohé její modifikace a také její zobecnění pro soustavy nelineárních rovnic. Užitečnost metod je demonstrována na různých příkladech. Na závěr jsou uvedeny Newton-Cotesovy kvadrurní formule pro numerické integrování.

Summary

Topic of this bachelor thesis are Newton's methods for numerical solutions of various problems. Especially the problems of solving nonlinear equations and systems of nonlinear equations, as well as numerical integration are explained. The Newton's method for solving nonlinear equations is presented, as well as its many modifications and its generalisation for systems of nonlinear equations. Usefulness of methods is demonstrated on various examples. In the end, Newton-Cotes quadrature formulae for numerical integration are presented.

Klíčová slova

numerické metody, nelineární rovnice, soustavy nelineárních rovnic, Newtonova metoda, metody Newtonova typu, numerické integrování, Newton-Cotesovy kvadrurní formule

Keywords

numerical methods, nonlinear equations, systems of nonlinear equations, Newton's method, Newton-type methods, numerical integration, Newton-Cotes quadrature formulae

Prohlašuji, že jsem bakalářskou práci *Newton a numerická matematika* vypracoval samostatně pod vedením Mgr. Jitky Zatočilové, Ph.D. s použitím materiálů uvedených v seznamu literatury.

Štěpán Obrátil

Rád bych poděkoval vedoucí práce Mgr. Jitce Zatočilové Ph.D. za podporu, vlídný přístup a nápomoc při vypracování této bakalářské práce.

Štěpán Obrátil

Obsah

1	Úvod	2
2	Nelineární rovnice	4
2.1	Řešení jedné nelineární rovnice	4
2.2	Startovací metody	5
2.2.1	Metoda bisekce	5
2.2.2	Metoda Regula falsi	7
2.3	Newtonova metoda a metody Newtonova typu	8
2.3.1	Newtonova metoda	8
2.3.2	Metody Newtonova typu s derivací	11
2.3.3	Metody Newtonova typu bez derivace	22
2.4	Srovnání Newtonovy metody a metod Newtonova typu	27
3	Soustavy nelineárních rovnic	32
3.1	Řešení soustav nelineárních rovnic	32
3.2	Newtonova metoda a metody Newtonova typu	32
3.2.1	Newtonova metoda	32
3.2.2	Metody Newtonova typu	33
3.3	Srovnání metod	38
4	Numerický výpočet integrálů	41
4.1	Newton-Cotesovy kvadrurní formule	41
4.2	Složené Newton-Cotesovy kvadrurní formule	46
4.3	Srovnání Newton-Cotesových kvadrurních formulí	48
5	Závěr	51
6	Seznam použitých zkratk a symbolů	55

1. Úvod

Tato bakalářská práce je zaměřena na dílo Isaaca Newtona v oblasti numerické matematiky. Sir Isaac Newton, narozen roku 1643 v Anglii, byl jeden z nejvýznamnějších vědců v historii. Za 84 let života publikoval nespočet převratných vědeckých prací, které posunuly technologický pokrok významně kupředu. Jeho přínos v oblasti fyziky, matematiky, astronomie, ale i alchymie či teologie je obrovský a má dosah do současnosti. Ne nadarmo bývá Isaac Newton označován za nejvlivnější osobu v dějinách lidstva.

Se jménem Isaac Newton se mnohým jistě vybaví především jeho objevy ze světa fyziky. Kromě zakladatelství teorie gravitace a klasické mechaniky je mu připisováno rovněž třeba autorství teorie barev, studium chování kapalin a především základy exaktní vědy jako takové.

V matematice je Newtonovo jméno spojováno především s objevem infinitesimálního počtu, o jehož autorství vedl dlouholetý spor s Gottfriedem Leibnizem. Dále pak třeba se zobecněním binomické věty, klasifikací kubických křivek v rovině, rozvojem mocninných řad a dalšími. Říká se, že významně rozvedl všechny tehdy studované obory matematiky.

V této práci nás však bude zajímat, především přínos Isaaca Newtona v oboru numerické matematiky. Známa Newtonova metoda je mocný nástroj pro numerické řešení jedné nelineární rovnice a je dodnes zkoumána a rozvíjena. Tato metoda je rovněž zobecněna pro soustavy nelineárních rovnic. Dále Newtonovo jméno nesou Newton-Cotesovy kvadrurní formule. Zásadně je Newtonovo jméno také zapsáno v oblasti optimalizace, interpolace a mnohých dalších. Jeho dílo je velmi rozsáhlé a vydalo by za víc, než je rozsah této bakalářské práce. Asi nejznámější je však Newtonova metoda pro řešení nelineární rovnice a proto jí budeme v této práci věnovat náležitou pozornost.

Práce je rozdělena na čtyři kapitoly a Newtonově metodě pro řešení jedné nelineární rovnice je věnována druhá kapitola. Jsou zde představeny základní metody řešení nelineárních rovnic, které zde označujeme jako startovací metody. Tyto metody sice nemají na první pohled s algoritmem Newtonovy metody nic společného, ale jsou velmi užitečné pro nalezení vhodné počáteční aproximace jak pro Newtonovu metodu, tak i pro její modifikace. Dále je ukázána samotná Newtonova metoda, její principy a vlastnosti. Velkou část této kapitoly pak zabírají různé modifikace Newtonovy metody, které zde dělíme na dva základní typy: ty, které ve svém algoritmu obsahují derivaci a na ty, které derivaci neobsahují. U některých metod je vysvětlen i princip jejich odvození, výhody a nevýhody. Na konci kapitoly jsou tyto metody mezi sebou porovnány na několika vybraných nelineárních rovnicích.

Třetí kapitola je zaměřena na řešení soustav nelineárních rovnic. Při řešení soustav nelineárních rovnic lze využít některé poznatky a principy z hledání řešení jedné nelineární rovnice. U soustav ale může být výrazně obtížnější nalezení vhodné počáteční aproximace, protože žádná z metod nemá zaručenu konvergenci. Kromě Newtonovy metody pro řešení soustav nelineárních rovnic v této kapitole opět uvádíme některé z jejich modifikací, které jsou pak srovnány s původní Newtonovou metodou na několika vybraných soustavách nelineárních rovnic.

Aby tato bakalářská práce nebyla zaměřena pouze na Newtonovy metody spojené s řešením jedné či více nelineárních rovnic, tak je poslední kapitola věnována Newtonově odkazu v numerickém integrálním počtu. Čtvrtá kapitola se tedy věnuje nejdříve obecně problému numerického výpočtu integrálů a potom konkrétně Newton-Cotesovým kvad-

raterním formulím. Jsou zde uvedeny základní otevřené i uzavřené kvadraturní formule, složené kvadraturní formule a jejich vlastnosti. Na konci jsou na několika příkladech porovnány výsledky získané použitím různých kvadraturních formulí.

2. Nelineární rovnice

Řada problémů řešených v aplikované matematice, optimalizaci, ekonomii, chemii a celé řadě různých technických odvětví (materiálové obory, pružnost a pevnost, mechanika plynů aj.) vede k problému řešení nelineárních rovnic. Mnohdy je však nalezení přesného řešení nemožné, nebo příliš náročné. V takovém případě přichází na řadu numerické metody, které naleznou řešení alespoň přibližně s požadovanou přesností. Tedy je zřejmé, že jsou numerické metody velmi významné pro řadu oborů a je důležité se jimi dále zabývat, zkoumat již fungující metody a zdokonalovat je.

V této bakalářské práci se budeme nejprve zabývat iteračními numerickými metodami pro řešení jedné nelineární rovnice $f(x) = 0$. Tyto metody z jedné nebo několika počátečních aproximací hledaného kořene ξ generují posloupnost x_0, x_1, x_2, \dots , která ke kořenu ξ konverguje. Mezi nejznámější metody patří Newtonova metoda, která je jednoduše implementovatelná a má dobré vlastnosti. Z této metody je pak odvozeno mnoho dalších metod, které se snaží zlepšit některé vlastnosti Newtonovy metody. Významnou roli u těchto metod hraje konvergence, která je závislá na vhodné počáteční aproximaci řešení. K hledání takové vhodné aproximace se používají metody, u kterých je konvergence zaručena. Takové metody zde nazýváme startovací metody a uvádíme je v kapitole 2.2.

2.1. Řešení jedné nelineární rovnice

Na začátek pro přehlednost uvedeme a vysvětlíme některé pojmy a definice, které budou dále užívány.

Nelineární rovnicí je zde myšlena rovnice ve tvaru

$$f(x) = 0, \quad (2.1)$$

kde x je reálná proměnná a $f(x)$ je spojitá funkce, která má tolik spojitých derivací kolik jich je v dané iteraci zapotřebí. Řešením nelineární rovnice je potom číslo $\xi \in \mathbb{R}$, pro které platí $f(\xi) = 0$. Všechna taková ξ nazýváme *kořeny nelineární rovnice* (2.1).

Tato práce se bude věnovat pouze metodám pro určení jednoduchého reálného kořene rovnice (2.1), tj. $f'(\xi) \neq 0$. Iterační numerická metoda je taková metoda, která generuje svým chodem posloupnost *aproximací kořene* $\{x_k\}_{k=0}^{\infty}$. Na počátku volíme *počáteční aproximaci* x_0 .

Definice 2.1.1. Nechť $\{x_k\}_{k=0}^{\infty}$ je posloupnost aproximací generovaná numerickou metodou a nechť $\lim_{k \rightarrow \infty} x_k = \xi$. Pak řekneme, že použitá iterační numerická metoda *konverguje*.

Pojem, který bude mít při vyšetřování iteračních metod zásadní úlohu, je řád konvergence iterační metody. Nejprve definujeme chybu k -té iterace vztahem $e_k = (\xi - x_k)$.

Definice 2.1.2. Nechť ξ je kořenem rovnice (2.1) a posloupnost reálných čísel $\{x_k\}_{k=0}^{\infty}$ konverguje ke ξ . Když existuje číslo $p \in \mathbb{R}^+$ a konstanta $c \neq 0$ taková, že

$$\lim_{k \rightarrow \infty} \frac{|\xi - x_{k+1}|}{|\xi - x_k|^p} = \lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^p} = c,$$

pak p se nazývá *řád konvergence* posloupnosti a c se nazývá asymptotická konstanta chyby, která závisí na funkci $f(x)$.

Řekneme, že daná metoda je *řádu* p , jestliže všechny konvergentní posloupnosti generované touto metodou mají řád konvergence větší nebo rovný p a nejméně jedna z těchto posloupností má řád konvergence rovný přesně p .

Také pokud platí chybová rovnice

$$|e_{k+1}| = c|e_k^p| + O(e_k^{p+1}),$$

tak p je řádem konvergence iterační metody.

Iterační metody bychom mohli navzájem porovnávat podle jejich rychlosti konvergence (tj. podle jejich řádů); velký smysl má však i porovnání jejich výpočtové pracnosti, která se měří množstvím výpočtů funkčních hodnot potřebných k dosažení dané přesnosti.

Definice 2.1.3. Nechť p je řád konvergence metody a r je počet vyhodnocení funkční hodnoty v jedné iteraci metody. Pak číslo $I_e = \sqrt[p]{p}$ nazveme *index efektivity* metody.

Řád konvergence a index efektivity jsou tedy základní jednoduché ukazatele, ze kterých poznáme, jak rychle a efektivně lze danou metodou dojít k řešení. Neplatí však jednoznačně, že čím jsou vyšší tyto hodnoty, tím je metoda lepší, neboť některé metody s vysokým řádem mohou mít jiná striktní omezení. Takže mohou sice být vysoce efektivní, ale jsou použitelné třeba jen pro vybranou skupinu funkcí. Některé metody s nižším řádem zase mohou být efektivnější v některých jiných specifických případech.

2.2. Startovací metody

Kořeny nelineární rovnice se obvykle hledají ve více krocích. Nejdříve kořeny separujeme, tedy najdeme intervaly, na kterých se nachází alespoň jeden kořen a z tohoto intervalu pak volíme počáteční aproximaci x_0 , příp. x_0 a x_1 v případě dvoubodové iterační metody. V druhém kroku aproximaci zpřesňujeme použitím metod, u kterých je zaručena konvergence. Všechny tyto metody mají konvergenci prvního řádu a jsou tedy velmi nepraktické k nalezení konečného řešení. Jejich nespornou výhodou je však to, že vždy konvergují. Pokud těmito metodami necháme spočítat několik iterací, vrátí nám výsledek, který sice pravděpodobně nedosahuje požadované přesnosti, ale je použitelný jako počáteční aproximace pro jiné, rychlejší metody. K separaci kořenů je užitečná následující věta.

Věta 2.2.1. [12] *Nechť funkce f je spojitá na intervalu $\langle a, b \rangle$ a platí $f(a)f(b) < 0$. Pak na intervalu $\langle a, b \rangle$ leží alespoň jeden kořen ξ funkce f .*

Nyní uvedeme alespoň dvě z těchto metod, kterými lze zlepšit počáteční aproximaci hledaného kořene nelineární rovnice (2.1).

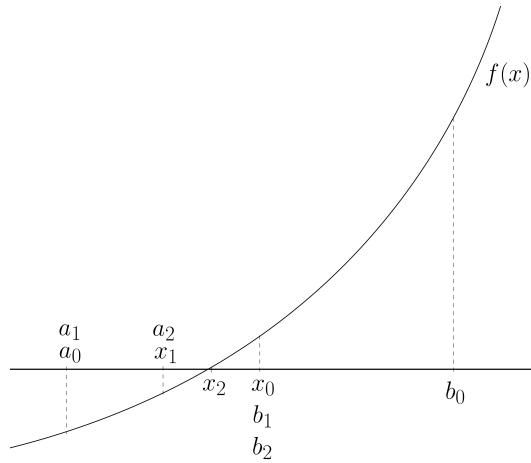
2.2.1. Metoda bisekce

Metoda bisekce, neboli metoda půlení intervalu, je velmi jednoduchá numerická metoda založená na větě (2.2.1). Nyní popíšeme algoritmus této metody.

2.2. STARTOVACÍ METODY

Nechť $f \in C\langle a, b \rangle$ a necht' $f(a)f(b) < 0$, tedy podle věty (2.2.1) leží na intervalu $\langle a, b \rangle$ alespoň jeden kořen nelineární rovnice (2.1). Položíme $a_0 = a, b_0 = b, x_0 = \frac{1}{2}(a_0 + b_0)$. Mohou nastat 3 možnosti:

- Pokud $f(x_0) = 0$, pak $x_0 = \xi$, tedy kořen je nalezen a algoritmus ukončíme.
- Pokud $f(a_0)f(x_0) < 0$, leží kořen v intervalu $\langle a_0, x_0 \rangle$, takže položíme $a_1 = a_0, b_1 = x_0$ a $x_1 = \frac{1}{2}(a_1 + b_1)$. Celý postup pak opakujeme pro nový interval $\langle a_1, b_1 \rangle$.
- Pokud $f(x_0)f(b_0) < 0$, leží kořen v intervalu $\langle x_0, b_0 \rangle$, takže položíme $a_1 = x_0, b_1 = b_0$ a $x_1 = \frac{1}{2}(a_1 + b_1)$. Celý postup pak opakujeme pro nový interval $\langle a_1, b_1 \rangle$.



Obrázek 2.1: metoda bisekce

Věta 2.2.2. (O konvergenci metody bisekce) *Nechť f je spojitá funkce na intervalu $\langle a, b \rangle$, $f(a)f(b) < 0$ a necht' f má na tomto intervalu jediný kořen ξ . Pak metoda bisekce generuje posloupnost $x_k = \frac{(a_k + b_k)}{2}$, $k = 0, 1, 2, \dots$, která konverguje ke kořenu ξ a aproximuje kořen ξ s přesností*

$$e_k \leq \frac{b - a}{2^{k+1}}$$

Důkaz. [12] Výše popsaným algoritmem metody dostáváme posloupnost intervalů

$$\langle a_0, b_0 \rangle \supset \langle a_1, b_1 \rangle \supset \dots \supset \langle a_k, b_k \rangle \supset \dots,$$

přičemž $f(a_n)f(b_n) < 0, n = 0, 1, \dots$

Pro koncové body těchto intervalů platí

$$a_0 \leq a_1 \leq \dots \leq a_k \leq \dots \leq \xi \leq \dots \leq b_k \leq \dots \leq b_1 \leq b_0$$

a délky těchto intervalů jsou dány vztahem

$$b_k - a_k = \frac{b_0 - a_0}{2^k}, \quad k = 1, 2, \dots,$$

kde $a_0 := a$ a $b_0 := b$.

Protože posloupnosti $\{a_k\}$, $\{b_k\}$ jsou omezené, monotonní a délka intervalů $\langle a_k, b_k \rangle$ konverguje k nule, platí

$$\lim_{k \rightarrow \infty} b_k = \lim_{k \rightarrow \infty} a_k = \xi.$$

Dále stačí ukázat, že ξ je kořenem rovnice $f(x) = 0$. Funkce f je spojitá a platí

$$f(a_k)f(b_k) < 0, \quad k = 0, 1, \dots$$

A tedy

$$\lim_{k \rightarrow \infty} f(b_k)f(a_k) = f^2(\xi) \leq 0,$$

ale odtud plyne, že $f(\xi) = 0$. Z tohoto postupu rovněž plyne, že

$$e_k \leq \frac{b-a}{2^{k+1}}, \quad x_k = \frac{a_k + b_k}{2},$$

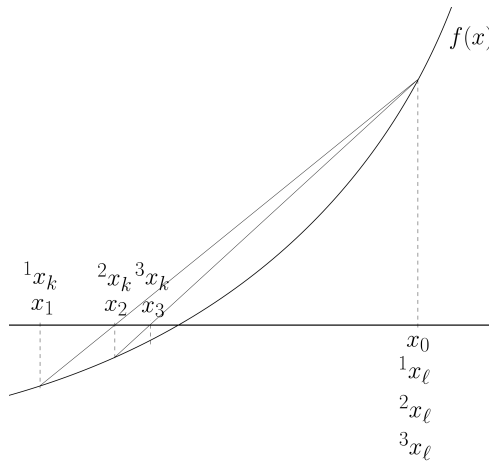
neboť $\xi \in \langle a_k, b_k \rangle$, $b_k - a_k = (b-a)/2^k$ a tedy $e_k \leq (b_k - a_k)/2 = (b-a)/2^{k+1}$. \square

2.2.2. Metoda Regula falsi

Metoda Regula falsi je další metodou vhodnou pro určení počáteční aproximace. Opět mějme interval $\langle a, b \rangle$ takový, že $f \in C\langle a, b \rangle$ a $f(a)f(b) < 0$, tedy na intervalu leží alespoň jeden kořen nelineární rovnice (2.1). Krajní body tohoto intervalu označíme jako počáteční aproximace, tedy položíme $x_0 = a$, $x_1 = b$. Další aproximaci kořene x_{k+1} tentokrát hledáme jako průsečík osy x s úsečkou spojující body $[x_k, f(x_k)]$ a $[x_\ell, f(x_\ell)]$, kde $\ell < k$ je největší index, pro který platí $f(x_\ell)f(x_k) < 0$. Výpočet aproximace tedy probíhá podle vzorce

$$x_{k+1} = x_k - f(x_k) \frac{x_k - x_\ell}{f(x_k) - f(x_\ell)}, \quad k = 0, 1, 2, \dots \quad (2.2)$$

Přitom pro $k = 1$ je $\ell = 0$. Dále se index ℓ určuje takto: pokud $f(x_{k+1})f(x_k) < 0$, pak $\ell = k$, v opačném případě se ℓ nemění. Princip metody je patrnější z obrázku (2.2).



Obrázek 2.2: Metoda Regula falsi

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

V následujících podkapitolách uvedeme kromě Newtonovy metody také některé její modifikace. Tyto modifikace budou v dalším textu rozděleny do dvou skupin: na ty, kde je v algoritmu užita první či druhá derivace zadané funkce (sem spadá i samotná Newtonova metoda) a na ty, kde je derivace nahrazena aproximací. V každé z těchto částí jsou metody tříděny podle způsobu jejich odvození a také řazeny podle řádu konvergence. Způsob odvození jednotlivých metod uvedeme, vzhledem k rozsahu bakalářské práce, jen u některých metod. U zbývajících bude uveden pouze odkaz na zdroj, kde lze o příslušné metodě získat bližší informace.

2.3. Newtonova metoda a metody Newtonova typu

Velká část metod uvedených v této kapitole nenese žádný oficiální název. Pro přehledné odlišení jednotlivých metod je však budeme nazývat podle autora, či autorů publikace, v níž byla daná metoda odvozena a popsána. Jednotlivé algoritmy popisující numerický výpočet dané metody pak pro přehlednost vzájemného srovnání označujeme zkratkou odvozenou z názvu metody, tedy např. Newtonovu metodu označíme jako *algoritmus NM*.

2.3.1. Newtonova metoda

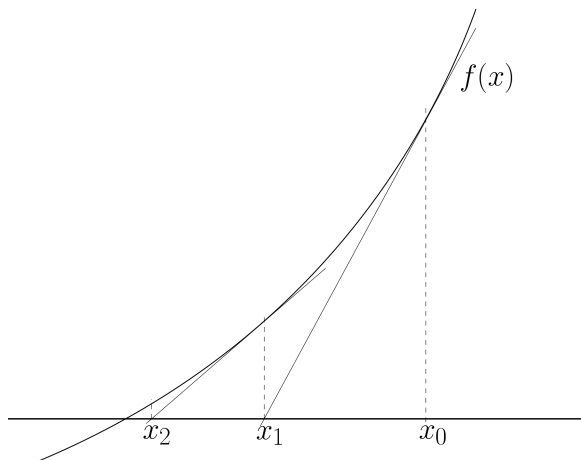
Newtonova metoda, nebo také Newton - Raphsonova metoda, nazvaná podle sira Isaaca Newtona, je asi nejznámější numerickou metodou pro řešení jedné nelineární rovnice. Mezi její výhody patří, že má konvergenci druhého řádu, tedy vyšší než startovací metody, ale na stranu druhou u této metody není zaručena konvergence, která závisí na vhodně zvolené počáteční aproximaci.

Tato metoda je též často nazývána metodou tečen, což plyne z geometrické interpretace výpočtového algoritmu metody (viz Obr. 2.3), který probíhá následovně: na počátku volíme aproximaci kořene x_0 . Dále sestrojíme tečnu ke grafu funkce $f(x)$ v bodě $[x_0, f(x_0)]$. Průsečík této tečny s osou x označíme x_1 . V bodě $[x_1, f(x_1)]$ sestrojíme znovu tečnu a její průsečík s osou x značíme x_2 . Takto postupujeme až získáme aproximaci x_k , která je kořenu ξ dostatečně blízká, což ověřujeme vhodně voleným kritériem pro ukončení algoritmu (viz kapitola 2.4).

Jeden krok metody tedy vyjádříme:

algoritmus NM:
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad k = 0, 1, 2, \dots \quad (2.3)$$

Ze zápisu metody lze spatřit další výhodu, kterou je nízká výpočtová náročnost. V každém kroku je potřeba spočítat pouze dvě funkční hodnoty. Tuto výhodu lze vyjádřit již zmíněným indexem efektivity $I_e = \sqrt{2} \doteq 1,4142$.



Obrázek 2.3: Newtonova metoda

Vzhledem k tomu, že konvergence metody je závislá na správné volbě počáteční aproximace x_0 , je vhodné algoritmus opatřit maximálním počtem povolených iterací. Pokud je tento počet překročen, je pravděpodobné, že metoda diverguje a je třeba zvolit jinou počáteční aproximaci. Vhodnou volbu počáteční aproximace x_0 a tím i konvergenci metody lze zajistit splněním podmínek následující věty.

Věta 2.3.1. (Fourierova podmínka) *Předpokládejme, že na intervalu $\langle a, b \rangle$ leží jediný kořen rovnice (2.1), a že $f'(x)$ a $f''(x)$ jsou spojité a nemění znaménko na $\langle a, b \rangle$. Pokud zvolíme počáteční aproximaci $x_0 \in \langle a, b \rangle$ tak, aby*

$$f(x_0) \cdot f''(x_0) > 0$$

pak Newtonova metoda konverguje.

Důkaz. [12] Budeme se zabývat případem $f(a) < 0, f(b) > 0, f''(x) \geq 0, \forall x \in \langle a, b \rangle$. V ostatních případech by důkaz probíhal podobně. Důkaz provedeme indukcí. Podle předpokladu věty zvolme $b = x_0$. Je třeba ukázat, že $\xi \leq x_1 \leq x_0$.

Jelikož

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)},$$

a $f(x_0) > 0, f'(x_0) > 0$, je zřejmé, že $x_0 > x_1$.

V dalším kroku použijeme Taylorův vzorec. Chybu v k -tém kroku značíme $|x_k - \xi| = e_k$.

$$0 = f(\xi) = f(x_0) + f'(x_0)e_0 + \frac{f''(\eta_0)}{2}e_0^2, \quad \eta_0 \in (\xi, x_0).$$

Protože $f''(\eta_0)e_0^2/2 \geq 0$, musí platit

$$f(x_0) + f'(x_0)(\xi - x_0) \leq 0$$

aby platila předchozí rovnost. Z toho dále plyne

$$\xi \leq x_0 - \frac{f(x_0)}{f'(x_0)} = x_1,$$

a tedy

$$\xi \leq x_1 \leq x_0.$$

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Nyní za předpokladu, že platí

$$\xi \leq x_k < x_{k-1} < \dots < x_0,$$

se stejným způsobem ukáže, že

$$\xi \leq x_{k+1} < x_k < \dots < x_0.$$

Posloupnost $\{x_k\}_{k=0}^{\infty}$ tedy konverguje k bodu ξ . □

Obecně však nemusí být snadné najít takovou vhodnou počáteční aproximaci. V takovém případě použijeme jednu z pomocných startovacích metod z předchozí kapitoly.

Věta 2.3.2. (O konvergenci Newtonovy metody) *Nechť je $f \in C^2 \langle a, b \rangle$. Nechť $\xi \in \langle a, b \rangle$ je kořenem rovnice $f(x) = 0$ a $f' \neq 0$. Dále nechť x_0 je vhodná počáteční aproximace. Pak pro posloupnost $\{x_k\}_{k=0}^{\infty}$ generovanou Newtonovou metodou platí*

a)

$$e_{k+1} \leq \frac{M}{2m} e_k^2, \quad (2.4)$$

b)

$$e_{k+1} \leq \frac{M}{2m} (x_{k+1} - x_k)^2, \quad (2.5)$$

kde $M = \max_{x \in I} |f''(x)|$, $m = \min_{x \in I} |f'(x)| > 0$ a I je interval obsahující ξ i x_0 .

Důkaz. [12] a) Nejdříve dokážeme vztah (2.4). Použijeme Taylorův vzorec ve tvaru

$$0 = f(\xi) = f(x_k) + f'(x_k)e_k + f''(\eta_k)\frac{e_k^2}{2},$$

kde bod η_k leží mezi body x_k a ξ .

Z Newtonovy metody plyne

$$x_{k+1}f'(x_k) = x_k f'(x_k) - f(x_k).$$

Dosazením do předchozího vztahu dostaneme

$$0 = -x_{k+1}f'(x_k) + \xi f'(x_k) + f''(\eta_k)\frac{e_k^2}{2}$$

a odtud

$$e_{k+1} = \frac{|f''(\eta_k)|}{2|f'(x_k)|} e_k^2 \leq \frac{M}{2m} e_k^2.$$

b) I pro důkaz druhého vztahu (2.5) použijeme Taylorova vzorce:

$$f(x_{k+1}) = f(x_k) + f'(x_k)(x_{k+1} - x_k) + \frac{f''(\alpha_k)}{2}(x_{k+1} - x_k)^2,$$

kde α_k leží mezi body x_k a x_{k+1} .

Opakujeme předchozí postup, tedy znovu z Newtonovy metody dostáváme

$$f(x_k) + f'(x_k)(x_{k+1} - x_k) = 0,$$

a tedy

$$f(x_{k+1}) = \frac{f''(\alpha_k)}{2}(x_{k+1} - x_k)^2.$$

Dále použitím věty o střední hodnotě dostaneme

$$f(x_{k+1}) = f(x_{k+1}) - f(\xi) = f'(\beta_k)e_{k+1},$$

kde bod β_k leží mezi body x_{k+1} a ξ . Dosazením do předchozího vztahu odtud plyne

$$e_{k+1} = \frac{|f''(\alpha_k)|}{2|f'(\beta_k)|}(x_{k+1} - x_k)^2,$$

a tedy i vztah (2.5). □

Tato věta nám říká, že chyba metody $e_k = (x_k - \xi)$ konverguje k nule, a tedy posloupnost $\{x_k\}_{k=0}^{\infty}$ konverguje ke kořeni ξ , pokud jsou splněny předpoklady věty. Protože chyba e_{k+1} je úměrná druhé mocnině chyby e_k , říkáme, že Newtonova metoda je druhého řádu.

2.3.2. Metody Newtonova typu s derivací

Dvojitá Newtonova metoda

Jako nejjednodušší způsob, jak zvýšit řád konvergence klasické Newtonovy metody se nabízí provádět dva její výpočtové kroky v rámci jednoho kroku. Taková metoda je pak vyjádřena následovně

$$\begin{aligned} \text{algoritmus DNM1:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)}, \\ x_{k+1} &= y_k - \frac{f(y_k)}{f'(y_k)}, \quad k = 0, 1, 2, \dots \end{aligned} \tag{2.6}$$

a je 4. řádu. Lehce však ověříme, že index efektivity je $I_e = \sqrt[4]{4} \doteq 1,4142$, tedy je stejný jako u Newtonovy metody.

Vhodnou úpravou můžeme dostat výpočtově trochu efektivnější podobu této metody. Když derivaci funkce budeme počítat jen jedenkrát za jeden krok algoritmu, dostaneme metodu

$$\begin{aligned} \text{algoritmus DNM2:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)}, \\ x_{k+1} &= y_k - \frac{f(y_k)}{f'(x_k)}, \quad k = 0, 1, 2, \dots \end{aligned} \tag{2.7}$$

Tato metoda je oproti předchozí sice pouze 3. řádu, ale kvůli menšímu počtu vyhodnocení funkce má lepší index efektivity $I_e = \sqrt[3]{3} \doteq 1,4422$, více viz [9].

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Mnoho numerických metod Newtonova typu je odvozeno ze základní věty integrálního počtu. Proto ji zde uvedeme:

Věta 2.3.3. *Nechť $f(x)$ je funkce definovaná na uzavřeném intervalu $\langle a, b \rangle$ s primitivní funkcí $F(x)$ na témž intervalu, tedy $F(x)$ a $f(x)$ jsou takové funkce, pro které platí*

$$f(x) = F'(x), \quad \forall x \in \langle a, b \rangle.$$

Je-li $f(x)$ na intervalu $\langle a, b \rangle$ Riemannovsky integrovatelná, pak platí

$$\int_a^b f(x) dx = F(b) - F(a). \quad (2.8)$$

Důkazem věty, ani vysvětlováním příslušných pojmů z integrálního počtu se v této práci nebudeme podrobněji zabývat. Vše potřebné lze nalézt např. v [13] nebo jiné literatuře zabývající se integrálním počtem funkcí jedné proměnné.

V dalším textu budeme při odvozování iteračních numerických metod potřebovat vztah (2.8) spíše v následujícím tvaru

$$\int_{x_k}^x f'(t) dt = f(x) - f(x_k), \quad (2.9)$$

kde pokud x je kořen nelineární rovnice $f(x) = 0$ a x_k je jeho aproximace, tak vztah (2.9) ještě můžeme upravit na

$$\int_{x_k}^x f'(t) dt + f(x_k) = 0. \quad (2.10)$$

Při odvozování nových iteračních metod je integrál ve vztahu (2.10) většinou nahrazován pomocí některé z následujících Newton-Cotesových kvadraturních formulí:

Levá obdélníková kvadraturní formule:

$$\int_a^b f'(t) dt \approx (b - a)f'(a), \quad (2.11)$$

Obdélníková kvadraturní formule vztažená k prostřednímu bodu intervalu (tzv. midpoint rule):

$$\int_a^b f'(t) dt \approx (b - a)f' \left(\frac{a}{2} + \frac{b}{2} \right). \quad (2.12)$$

Lichoběžníková kvadraturní formule:

$$\int_a^b f'(t) dt \approx (b - a) \frac{(f'(a) + f'(b))}{2}. \quad (2.13)$$

Poznámka. Více se budeme Newton-Cotesovým kvadraturním formulím věnovat v kapitole 4.

Užijeme-li pro aproximaci integrálu ve vztahu (2.10) levou obdélníkovou kvadrurní formuli ve tvaru

$$\int_{x_k}^x f'(t) dt \approx (x - x_k)f'(x_k), \quad (2.14)$$

kde místo x budeme uvažovat další spočtenou aproximaci kořene x_{k+1} , tak dostáváme klasickou Newtonovu metodu (2.3).

Noorovy metody

Noor a spol. v [22] při odvození nové metody použili lichoběžníkovou kvadrurní formuli (2.13) ve tvaru

$$\int_{x_k}^x f'(t) dt \approx (x - x_k) \frac{(f'(x_k) + f'(x))}{2}. \quad (2.15)$$

Dosazením (2.15) do (2.10) dostáváme následující vztah

$$f(x_k) + \frac{x - x_k}{2}(f'(x) + f'(x_k)) = 0, \quad (2.16)$$

který lze dále upravit na

$$x = x_k - \frac{2f(x_k)}{f'(x_k)} - (x - x_k) \frac{f'(x)}{f'(x_k)}.$$

To už je tvar, ze kterého lze snadno vyjádřit novou iterační metodu, pokud také místo x uvažujeme další spočtenou aproximaci kořene x_{k+1} :

$$x_{k+1} = x_k - \frac{2f(x_k)}{f'(x_k)} - (x_{k+1} - x_k) \frac{f'(x_{k+1})}{f'(x_k)}. \quad (2.17)$$

Metoda daná vztahem (2.17) je však implicitní, tedy pro výpočet x_{k+1} potřebuje znát hodnotu x_{k+1} a $f'(x_{k+1})$ na pravé straně rovnice. Tento problém lze vyřešit použitím tzv. metody prediktor - korektor. Nejdříve v algoritmu jako prediktor použijeme Newtonovu metodu, tedy spočteme předběžnou aproximaci kořene x_{k+1} algoritmem (2.3), kterou označíme jako y_k . Jako korektor pak vezmeme novou metodu (2.17) a dostaneme

$$\begin{aligned} \text{algoritmus NoM1:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\ x_{k+1} &= x_k - \frac{2f(x_k)}{f'(x_k)} - (y_k - x_k) \frac{f'(y_k)}{f'(x_k)}, \quad k = 0, 1, 2, \dots \end{aligned} \quad (2.18)$$

Prostým přeuspořádáním vztahu (2.18) můžeme dostat jinou variantu této metody ve tvaru

$$\text{algoritmus NoM2:} \quad x_{k+1} = y_k - \frac{f(x_k)}{f'(x_k)} + \frac{f'(y_k)f(x_k)}{f'^2(x_k)}, \quad k = 0, 1, 2, \dots \quad (2.19)$$

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Uvedené Noorovy metody (2.18) a (2.19) jsou 2. řádu a jejich index efektivity je $I_e = \sqrt[3]{2} \doteq 1,2599$, tedy mají nižší index efektivity než Newtonova metoda a přitom stejný řád. Přesto, jak je vidět v tabulce srovnání metod (2.3), mohou pro některé typy nelineárních rovnic dávat lepší výsledky než Newtonova metoda. Příslušné důkazy řádů konvergence těchto metod lze nalézt v [22].

Weerakoon - Fernandova metoda

Vztah (2.16) může být upraven ještě na jiný tvar, např.

$$x = x_k - \frac{2f(x_k)}{f'(x_k) + f'(x)},$$

z čehož obdobně, jako v předchozích případech, dostáváme

$$\text{algoritmus WFM:} \quad x_{k+1} = x_k - \frac{2f(x_k)}{f'(x_k) + f'(y_k)}, \quad k = 0, 1, 2, \dots, \quad (2.20)$$

kde $y_k = x_k - f(x_k)/f'(x_k)$.

Tuto metodu odvodili Weerakoon a Fernando a ve svém článku [31] ukázali, že tato metoda má konvergenci 3. řádu. Spočteme-li index efektivity této metody $I_e = \sqrt[3]{3} \doteq 1,4422$, tak zjistíme, že i ten je nepatrně vyšší než u Newtonovy metody.

Halleyho metoda

Použitím Taylorova rozvoje funkce $f'(y_k)$ ve tvaru

$$\begin{aligned} f'(y_k) &\simeq f'(x_k) + f''(x_k)(y_k - x_k) = \\ &= f'(x_k) - f''(x_k) \frac{f(x_k)}{f'(x_k)}, \end{aligned} \quad (2.21)$$

a jeho dosazením do (2.20) lze odvodit další metodu

$$\text{algoritmus HaM:} \quad x_{k+1} = x_k - \frac{2f(x_k)f'(x_k)}{2f'^2(x_k) - f(x_k)f''(x_k)}, \quad k = 0, 1, 2, \dots \quad (2.22)$$

Ta se nazývá po svém autorovi Halleyho metoda a je 3. řádu, viz [16], s indexem efektivity $I_e = \sqrt[3]{3} \doteq 1,4422$. Halleyho metoda (2.22) však obsahuje druhou derivaci funkce a proto může být výpočtově náročnější než předchozí Weerakoon - Fernandova metoda (2.20) i přesto, že je stejného řádu a má i stejný index efektivity jako tato metoda.

Frontini - Sormaniova metoda

Obdobným způsobem, jako u předchozích metod, postupovali Frontini a Sormani, kteří odvodili další metodu 3. řádu použitím obdélníkové kvadraturní formule vztažené k prostřednímu bodu intervalu (2.12) ve tvaru

$$\int_{x_k}^x f'(t) dt \approx (x - x_k)f' \left(\frac{x_k + x}{2} \right). \quad (2.23)$$

Jejím dosazením do (2.10) dostáváme metodu ve tvaru

$$x_{k+1} = x_k - \frac{f(x_k)}{f' \left(\frac{x_k + x_{k+1}}{2} \right)},$$

ve které když x_{k+1} v pravé straně rovnice nahradíme Newtonovou metodou (2.3), získáme

algoritmus FSM:
$$x_{k+1} = x_k - \frac{f(x_k)}{f' \left(x_k - \frac{f(x_k)}{2f'(x_k)} \right)}, \quad k = 0, 1, 2, \dots \quad (2.24)$$

Index efektivity je $I_e = \sqrt[3]{3} \doteq 1,4422$. Takže se jedná o metodu stejného řádu a efektivity jako jsou předchozí metody WFM a HaM. Důkaz o konvergenci 3. řádu této metody lze nalézt v [7].

Homeierovy metody

Homeier ve svém článku [11] ukázal, že další možností je základní větu integrálního počtu (2.9) uvažovat nikoliv pro funkci $y = f(x)$, ale pro inverzní funkci $x(y)$, tedy

$$x(y) = x(y_k) + \int_{y_k}^y x'(t) dt. \quad (2.25)$$

Pro aproximaci integrálu v (2.25) byla použita obecná interpolační kvadraturní formule

$$Q_m(f) = (y - y_k) \sum_{j=1}^m w_j f(g_j)$$

kde $g_j = y_k + \tau_j(y - y_k)$, τ_j jsou uzly v intervalu $\langle 0, 1 \rangle$ a w_j jsou váhy. Ty jsou dány vztahy

$$\sum_{j=1}^m w_j = 1, \quad \sum_{j=1}^m w_j \tau_j = 1/2$$

tak, aby Q_m byla alespoň prvního řádu.

Aplikací této kvadraturní formule ve vztahu (2.25), dále užitím vztahů $y = y^* = y(x^*) = 0$, $x(y) = x^*$ a nahrazením $y_k = f(x_k)$ ekvivalentním vztahem $x(y_k) = x_k$ dostaneme rovnici

$$x^* = x_k - y_k \sum_{j=1}^m w_j x'((1 - \tau_j)y_k).$$

Jelikož

$$x'(y) = \left[\frac{df(x)}{dx} \Big|_{x(y)} \right]^{-1},$$

je potřeba určit body $x_{k,j} = x((1 - \tau_j)y_k)$. K tomu se předpokládá lineární závislost x a y v okolí x^* ve tvaru $y = f(x_k) - (x - x_k)y'_k$, kde $y'_k = f'(x_k)$. Z toho pak máme, že

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

$x_{k,j} = x_k - \tau_j y_k / y'_k$. Shrnutím všech předchozích poznatků dohromady dostaneme obecný zápis Homeierovy metody ve tvaru

$$x_{k+1} = x_k - f(x_k) \sum_{j=1}^m w_j \frac{1}{f'(x_k - \tau_j f(x_k) / f'(x_k))}, \quad k = 0, 1, 2, \dots \quad (2.26)$$

kde w_j jsou váhy a $\tau_j \in \langle 0, 1 \rangle$ uzly m -bodové interpolační kvadraturní formule. Vztah (2.26) tedy popisuje celou třídu metod, lišící se volbou m, w_j a τ_j . Každá metoda z uvedené třídy má konvergenci 3. řádu a index efektivity $I_e = \sqrt[3]{3} \doteq 1,4422$, viz [11].

Pokud bychom zvolili za $m = 1, w_1 = 1, \tau_1 = \frac{1}{2}$, což jsou koeficienty odpovídající obdélníkové kvadraturní formuli vztažené k prostřednímu bodu intervalu, tak dostaneme již uvedenou Frontini - Sormaniovu metodu (2.24).

Volbou koeficientů $m = 2, w_1 = w_2 = \frac{1}{2}, \tau_1 = 0, \tau_2 = 1$, které odpovídají lichoběžníkové kvadraturní formuli, dostáváme Homeierovu metodu:

algoritmus HoM:
$$x_{k+1} = x_k - \frac{f(x_k)}{2} \left(\frac{1}{f'(x_k)} + \frac{1}{f'(x_k - f(x_k) / f'(x_k))} \right),$$

$$k = 0, 1, 2, \dots \quad (2.27)$$

Kouovy metody

Pro zvýšení řádu konvergence použil Kou v [17] při odvození svých metod již existující metody 3. řádu, které začlenil do dalšího algoritmu. Tím vytvořil třídu metod 5. řádu, která využívá již popsanych metod (2.20), (2.24) a (2.27).

Aproximaci kořene spočtenou těmito metodami 3. řádu pro jednoduchost označíme jako $u_k := g_3(x_k)$ a $y_k := x_k - f(x_k) / f'(x_k)$ je opět aproximace spočtená Newtonovou metodou.

Základní věta integrálního počtu (2.9) je nyní uvažována ve tvaru:

$$f(x) = f(u_k) + \int_{u_k}^x f'(t) dt.$$

Použitím obdélníkové kvadraturní formule (2.11) v předchozím vztahu dostáváme první Kouovu třídu iteračních metod

$$x_{k+1} = u_k - \frac{f(u_k)}{f'(y_k)}, \quad k = 0, 1, 2, \dots \quad (2.28)$$

kde u_k a y_k odpovídají popisu výše.

Druhou Kouovu třídu metod odvodíme, pokud vyjádříme funkce $f'(y_k)$ a $f'(\frac{1}{2}(x_k + y_k))$ pomocí Taylorova rozvoje

$$f'(y_k) \simeq f'(x_k) + f''(x_k)(y_k - x_k),$$

$$f'(\tfrac{1}{2}(x_k + y_k)) \simeq f'(x_k) + \tfrac{1}{2}f''(x_k)(y_k - x_k),$$

a dvojnásobek druhé rovnice odečteme od první. Tak získáme aproximaci $f'(y_k)$ ve tvaru

$$f'(y_k) \simeq 2f'(\tfrac{1}{2}(x_k + y_k)) - f'(x_k),$$

kteřou dosadíme do (2.28) a máme tak třídu metod

$$x_{k+1} = u_k - \frac{f(u_k)}{2f'(\tfrac{1}{2}(x_k + y_k)) - f'(x_k)}, \quad k = 0, 1, 2, \dots \quad (2.29)$$

Kou takto tedy odvodil dvě třídy metod (2.28) a (2.29) používající ve svém algoritmu tři různé metody 3. řádu, tedy celkem šest možných kombinací. Ty se od sebe ale liší především výpočtovou náročností. My jsme z nich vybrali pouze následující tři nejméně náročné kombinace.

Použitím Weerakoon - Fernandovy metody (2.20) a třídy (2.28), máme metodu

$$\begin{aligned} \text{algoritmus KoM1:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\ u_k &= x_k - \frac{2f(x_k)}{f'(x_k) + f'(y_k)} \\ x_{k+1} &= u_k - \frac{f(u_k)}{f'(y_k)}, \quad k = 0, 1, 2, \dots \end{aligned} \quad (2.30)$$

Dále kombinací Frontini - Sormaniovy metody (2.24) a třídy (2.29) dostaneme

$$\begin{aligned} \text{algoritmus KoM2:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\ u_k &= x_k - \frac{f(x_k)}{f'(\tfrac{1}{2}(x_k + y_k))} \\ x_{k+1} &= u_k - \frac{f(u_k)}{2f'(\tfrac{1}{2}(x_k + y_k)) - f'(x_k)}, \quad k = 0, 1, 2, \dots \end{aligned} \quad (2.31)$$

Nakonec, Homeierova metoda (2.27) s třídou (2.28) dává

$$\begin{aligned} \text{algoritmus KoM3:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\ u_k &= x_k - \frac{f(x_k)}{2} \left(\frac{1}{f'(x_k)} + \frac{1}{f'(y_k)} \right) \\ x_{k+1} &= u_k - \frac{f(u_k)}{f'(y_k)}, \quad k = 0, 1, 2, \dots \end{aligned} \quad (2.32)$$

Všechny tyto metody jsou 5. řádu a mají i shodný index efektivity $I_e = \sqrt[4]{5} \doteq 1,4953$. Důkaz o řádu konvergence těchto metod lze nalézt v [17].

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Jak jsme se již zmínili na začátku této kapitoly, tak z důvodu rozsahu bakalářské práce již nebudeme u dalších metod Newtonova typu s derivací uvádět způsob jejich odvození, ale omezíme se pouze na uvedení jejich algoritmu, řádu konvergence, indexu efektivity a odkazu na literaturu, kde se lze o dané metodě dozvědět větší podrobnosti.

Chunovy metody 4. řádu

Obecný algoritmus metod:

$$x_{k+1} = x_k + (1 + \beta) \frac{f(x_k) + f(y_k)}{f'(x_k)} - 2 \frac{f^2(x_k)}{f'(x_k)(f(x_k) - f(y_k))} - \beta \left(\frac{f(x_k)}{f'(x_k)} + \frac{f'(x_k)f(y_k)}{f^2(x_k) + f'^2(x_k)} \right), \quad k = 0, 1, 2, \dots \quad (2.33)$$

kde opět $y_k = x_k - f(x_k)/f'(x_k)$. Různou volbou konstanty β byly získány následující metody:

Pro $\beta = -1$:

$$\text{algoritmus CM2:} \quad x_{k+1} = x_k - \frac{2f^2(x_k)}{f'(x_k)(f(x_k) - f(y_k))} + \frac{f(x_k)}{f'(x_k)} + \frac{f'(x_k)f(y_k)}{f^2(x_k) + f'^2(x_k)}, \quad k = 0, 1, 2, \dots \quad (2.34)$$

a pro $\beta = -2$:

$$\text{algoritmus CM3:} \quad x_{k+1} = x_k - \frac{f(x_k) + f(y_k)}{f'(x_k)} + 2 \left(\frac{f(x_k)}{f'(x_k)} - \frac{f^2(x_k)}{f'(x_k)(f(x_k) - f(y_k))} + \frac{f'(x_k)f(y_k)}{f^2(x_k) + f'^2(x_k)} \right), \quad k = 0, 1, 2, \dots \quad (2.35)$$

Volbou $\beta = 0$:

$$\text{algoritmus KoM4:} \quad x_{k+1} = x_k - \frac{f^2(x_k) + f^2(y_k)}{f'(x_k)(f(x_k) - f(y_k))}, \quad k = 0, 1, 2, \dots \quad (2.36)$$

Tuto metodu již dříve odvodili Kou a spol. v [18] a proto má odlišný název.

Všechny tyto metody jsou 4. řádu a mají i stejný index efektivity $I_e = \sqrt[3]{4} \doteq 1,5874$. Více lze nalézt v [14].

Potra - Ptáková metoda

Řád konvergence této metody je 3 a index efektivity metody je $I_e = \sqrt[3]{3} \doteq 1,4422$. Podrobné odvození a důkaz konvergence metody lze najít v [23]. U této metody bychom ještě poznamenali, že byla po dlouhou dobu jedinou známou metodou s konvergencí 3. řádu, která neobsahovala derivace vyšších řádů.

algoritmus PPM:
$$x_{k+1} = x_k - \frac{f(x_k) + f\left(x_k - \frac{f(x_k)}{f'(x_k)}\right)}{f'(x_k)}, \quad k = 0, 1, 2, \dots$$
 (2.37)

Newton - Steffensenova metoda

Tato metoda 3. řádu s indexem efektivity $I_e = \sqrt[3]{3} \doteq 1,4422$ je blíže popsána např. v [26] a má předpis ve tvaru

algoritmus NSM:
$$y_k = x_k - \frac{f(x_k)}{f'(x_k)},$$

$$x_{k+1} = x_k - \frac{f^2(x_k)}{f'(x_k)(f(x_k) - f(y_k))}, \quad k = 0, 1, 2, \dots$$
 (2.38)

Chunova metoda 3. řádu

Metoda je podrobně odvozena v [15]. Je 3. řádu, index efektivity metody je $I_e = \sqrt[3]{3} \doteq 1,4422$ a předpis má ve tvaru

algoritmus CM1:
$$y_k = x_k - \frac{f(x_k)}{f'(x_k)},$$

$$x_{k+1} = x_k - \left(\frac{f(x_k)}{f'(x_k)} + \frac{f'(x_k)f(y_k)}{f^2(x_k) + f'^2(x_k)} \right), \quad k = 0, 1, 2, \dots$$
 (2.39)

Jarrattova metoda

Jarrattova metoda je popsána např. v [1], je 4. řádu a má index efektivity $I_e = \sqrt[4]{4} \doteq 1,5874$. Metoda je tvaru

algoritmus JM:
$$y_k = x_k - \frac{2f(x_k)}{3f'(x_k)},$$

$$x_{k+1} = x_k - \left(1 - \frac{3}{2} \frac{f'(y_k) - f'(x_k)}{3f'(y_k) - f'(x_k)} \right) \frac{f(x_k)}{f'(x_k)}, \quad k = 0, 1, 2, \dots$$
 (2.40)

Fangova metoda

Fang a spol. odvodili metodu 5. řádu, která má index efektivity $I_e = \sqrt[4]{5} \doteq 1,4953$ a je tvaru

algoritmus FM:
$$y_k = x_k - \frac{f(x_k)}{f'(x_k)},$$

$$x_{k+1} = y_k - \frac{5f'^2(x_k) + 3f'^2(y_k)}{f'^2(x_k) + 7f'^2(y_k)} \cdot \frac{f(y_k)}{f'(x_k)}, \quad k = 0, 1, 2, \dots$$
 (2.41)

Více lze nalézt v [6].

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Kumarova metoda 5. řádu

Další metodu 5. řádu odvodili Kumar a spol.:

$$\begin{aligned}
 \text{algoritmus KuM1:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\
 z_k &= x_k - \frac{f(x_k) + f(y_k)}{f'(x_k)} \\
 x_{k+1} &= z_k - \frac{f(z_k)}{f'(y_k)}, \quad k = 0, 1, 2, \dots
 \end{aligned} \tag{2.42}$$

Její index efektivity je pouze $I_e = \sqrt[5]{5} \doteq 1,3797$, tedy velmi nízký. Tato metoda je rozvíjena v další kapitole, kdy z ní lze odvodit metodu 5. řádu bez derivací, která má už index efektivity vyšší. Bližší informace o metodě lze nalézt např. v [20].

Kumarova metoda 9. řádu

Zde je uvedena další Kumarova metoda, tentokrát 9. řádu. Její předpis je

$$\begin{aligned}
 \text{algoritmus KuM2:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\
 z_k &= y_k - \left[1 + \left(\frac{f(y_k)}{f(x_k)} \right)^2 \right] \frac{f(y_k)}{f'(y_k)} \\
 x_{k+1} &= z_k - \left[1 + 2 \left(\frac{f(y_k)}{f(x_k)} \right)^2 - 4 \left(\frac{f(z_k)}{f(y_k)} \right) \right] \frac{f(z_k)}{f'(y_k)}, \\
 k &= 0, 1, 2, \dots
 \end{aligned} \tag{2.43}$$

Tato metoda má index efektivity $I_e = \sqrt[5]{9} \doteq 1,5518$. Ale podobně jako předchozí Kumarovu metodu (2.42), i tuto metodu lze vylepšit odstraněním derivací. Důkaz a odvození lze nalézt v [19].

Džuničova metoda

Džunič a spol. vytvořili metodu 8. řádu ve tvaru

$$\begin{aligned}
 \text{algoritmus DM:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)}, \\
 z_k &= y_k - (1 - 2t)^{-1} \frac{f(y_k)}{f'(x_k)}, \\
 t &= \frac{f(y_k)}{f(x_k)}, \quad v = \frac{f(z_k)}{f(y_k)}, \quad w = tv = \frac{f(z_k)}{f(x_k)}, \\
 x_{k+1} &= z_k - \left(\frac{(1+v)(1+2w)}{(1-2t-t^2)} \right) \frac{f(z_k)}{f'(z_k)}, \quad k = 0, 1, 2, \dots
 \end{aligned} \tag{2.44}$$

Metoda má ve svém algoritmu pouze 4 vyhodnocení funkční hodnoty funkce, a tedy index efektivity $I_e = \sqrt[4]{8} \doteq 1,6818$ už je velmi vysoký oproti metodám nižších řádů. Důkaz lze nalézt v [4].

Sharmova metoda

Třída metod 8. řádu vytvořená Sharmou a spol. má rovněž index efektivity je $I_e = \sqrt[4]{8} \doteq 1,6818$ a je tvaru

$$\text{algoritmus ShM:} \quad x_{k+1} = z_k - (1 + w + \beta w^2) \left(\frac{f(z_k)f[x_k, y_k]}{f[y_k, z_k]f[x_k, z_k]} \right), \quad k = 0, 1, 2, \dots \quad (2.45)$$

kde

$$f[*, \circ] = \frac{f(\circ) - f(*)}{\circ - *}, \quad * \neq \circ, \quad (2.46)$$

y_k, z_k, t a w jsou dány podle vztahů v (2.44), a $\beta \in \mathbb{R}$ je volitelné číslo, kterým určíme podobu konkrétní metody z třídy. V této práci uvažujeme při numerickém srovnávání metod konkrétní metodu, kdy $\beta = 0$. Pro více informací a důkaz konvergence viz [27].

Wangova metoda

Wang a spol. odvodili metodu, která je rovněž 8. řádu s indexem efektivity $I_e = \sqrt[4]{8} \doteq 1,6818$:

$$\text{algoritmus WM:} \quad x_{k+1} = z_k - \frac{f(z_k)}{2f[x_k, z_k] + f[y_k, z_k] - 2f[x_k, y_k] + (y_k - z_k)f[y_k, x_k, x_k]}, \quad k = 0, 1, 2, \dots \quad (2.47)$$

kde, y_k a z_k jsou dány ze vztahů (2.44), $f[*, \circ]$ je dáno vztahem (2.46) a

$$f[y_k, x_k, x_k] = \frac{f[y_k, x_k] - f'(x_k)}{y_k - x_k}.$$

Více informací lze nalézt v [30].

Thukralova metoda 8. řádu

Metoda odvozená Thukralem a spol. je tvaru

$$\begin{aligned} \text{algoritmus TM1:} \quad y_k &= x_k - \frac{f(x_k)}{f'(x_k)}, \\ z_k &= y_k - K(t) \frac{f(y_k)}{f'(x_k)}, \\ x_{k+1} &= z_k - \left(\frac{(1 + K(t))^2}{4} + \frac{v}{1 - bv} + \frac{4w}{1 - cw} + pt^4 \right) \frac{f(z_k)}{f'(x_k)}, \\ k &= 0, 1, 2, \dots \end{aligned} \quad (2.48)$$

kde

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

$$K(t) = \frac{1 + at}{1 + (a - 2)t}, \quad (2.49)$$

t, v a w jsou dány ze vztahů (2.44), a $a, b, c, p \in \mathbb{R}$ jsou volitelné konstanty. Při numerickém srovnávání metod uvažujeme konstanty $a = b = c = 0, p = 3$. Metoda je 8. řádu a má rovněž jako předcházející metody 8. řádu index efektivity $I_e = \sqrt[4]{8} \doteq 1,6818$. Pro více informací viz [29].

Thukralova metoda 9. řádu

Velmi silnou metodu 9. řádu vyvinul Thukral a spol.. Krom vysokého řádu má i vysoký index efektivity $I_e = \sqrt[4]{9} \doteq 1,7132$. Metoda je popsána v [28]. Je definována vztahy

$$\text{algoritmus TM2:} \quad y_k = x_k - \frac{f(x_k)}{f'(x_k)} - (2x_k)^{-1} \left[x_k^2 - \left(\frac{f(x_k)}{f'(x_k)} \right)^2 \right], \quad (2.50)$$

$$x_{k+1} = y_k - \frac{f(y_k)}{f'(y_k)} - (2y_k)^{-1} \left[y_k^2 - \left(\frac{f(y_k)}{f'(y_k)} \right)^2 \right], \quad k = 0, 1, 2, \dots \quad (2.51)$$

Pokud bychom uvažovali pouze vztah (2.50), dostali bychom metodu třetího řádu. Prostým opakováním tohoto kroku v rámci jedné iterace pak dostáváme metodu devátého řádu. Tedy jedná se o stejný postup, jako při vytvoření Dvojité Newtonovy metody (2.6). Nepochybnou nevýhodou této metody je fakt, že ji lze použít pouze pro řešení nelineárních rovnic s nulovým kořenem (viz kapitola 2.4).

2.3.3. Metody Newtonova typu bez derivace

Newtonova metoda a další metody z ní v tomto textu dosud odvozené používají ve svém algoritmu výpočet derivace funkce. Někdy ale může být obtížné, nebo výpočetně náročné derivaci počítat. Proto byly z Newtonovy metody odvozeny i takové numerické metody, které místo derivace funkce používají pouze její aproximaci pomocí difference, což může vést v mnoha případech k významnému zvýšení indexu efektivity. Avšak i méně efektivní metody používající diferenci místo derivace mohou být užitečné pro konkrétní případy.

Nejprve uvedeme metody s diferencí, které byly odvozeny tak, že do předpisu některé "Newtonovské metody" byla přímo dosazena na místo derivace její aproximace.

Metoda sečen

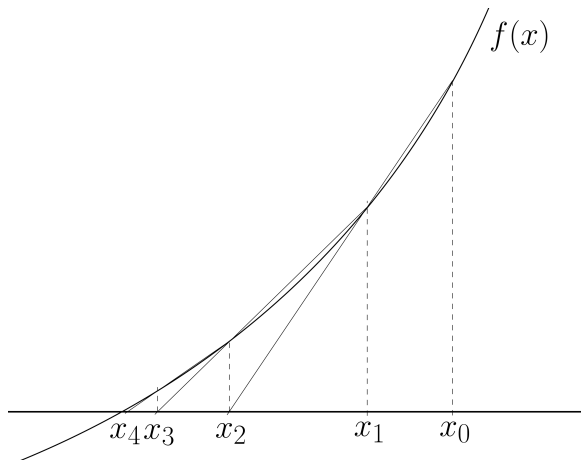
Pokud derivaci funkce $f(x)$ v bodě x_k aproximujeme zpětnou diferencí

$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} \quad (2.52)$$

a tuto aproximaci dosadíme přímo do předpisu Newtonovy metody (2.3), tak dostaneme algoritmus metody sečen:

$$\text{algoritmus MS:} \quad x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k), \quad k = 0, 1, 2, \dots \quad (2.53)$$

Název této metody pochází z její geometrické interpretace. Novou aproximaci x_{k+1} nalezneme jako průsečík osy x a sečny procházející body $[x_k, f(x_k)]$ a $[x_{k-1}, f(x_{k-1})]$, viz Obr. 2.4 níže.



Obrázek 2.4: Metoda sečen

Řád konvergence metody sečen je $\frac{1+\sqrt{5}}{2} \doteq 1,618$ (důkaz lze nalézt např. v [10]) a index efektivity je $I_e = \sqrt[3]{1,618} = 1,2720$, obě hodnoty jsou tedy nižší než u Newtonovy metody. Pokud je však výpočet derivace náročný, může být tato metoda výhodnější. Oproti Newtonově metodě jsou však u metody sečen potřeba dvě počáteční aproximace x_0 a x_1 (tedy řekneme, že metoda je dvoubodová). Tyto počáteční aproximace je třeba i zde volit velmi pečlivě, protože na nich závisí konvergence či divergence metody. I když by mohla být metoda sečen v některých případech srovnatelná s Newtonovou metodou, tak ji ve srovnání metod v kap. 2.4 neuvádíme právě z důvodu rozdílnosti počtu počátečních aproximací.

Steffensenova metoda

Steffensenova metoda je spolu s metodou sečen základní metodou pro výpočet kořenů nelineární rovnice, která nepoužívá derivaci funkce. Metodu odvodíme podobným způsobem jako metodu sečen. Derivaci funkce aproximujeme dopřednou diferencí

$$f'(x_k) \approx \frac{f(x_k + f(x_k)) - f(x_k)}{f(x_k)}. \quad (2.54)$$

kde $f(x_k)$ je zvolená délka kroku dopředné difference.

Tuto aproximaci dosadíme do Newtonovy metody (2.3) a dostaneme Steffensenovu metodu

algoritmus StM:
$$x_{k+1} = x_k - \frac{f^2(x_k)}{f(x_k + f(x_k)) - f(x_k)}, \quad k = 0, 1, 2, \dots \quad (2.55)$$

Tato metoda je rovněž jako Newtonova metoda druhého řádu a má i stejný index efektivity. Její výhoda tedy spočívá především v absenci derivace funkce v algoritmu metody. Také oproti metodě sečen má Steffensenova metoda tu výhodu, že stačí na počátku znát pouze jednu aproximaci x_0 kořene ξ , čímž se stává srovnatelnější s Newtonovou metodou. Důkaz o řádu konvergence metody lze nalézt např. v [16].

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Kumarova metoda 5. řádu bez derivace

Také z Kumarovy metody 5. řádu (2.42), kterou jsme uvedli mezi metodami, které mají ve svém algoritmu derivaci funkce, lze získat novou metodu bez derivace, nahradíme-li derivaci funkce dopřednou diferencí (2.54). Kumar ve svém článku [20] používá stručnější zápis této aproximace ve tvaru

$$f'(x_k) \approx f[x_k, w_k] = \frac{f(w_k) - f(x_k)}{w_k - x_k} = \frac{f(w_k) - f(x_k)}{f(x_k)}, \quad (2.56)$$

kde $w_k = x_k + f(x_k)$, a obdobně aproximuje i

$$f'(y_k) \approx \frac{f[y_k, x_k]f[y_k, w_k]}{f[x_k, w_k]}. \quad (2.57)$$

kde

$$f[*, \circ] = \frac{f(\circ) - f(*)}{\circ - *}, \quad * \neq \circ. \quad (2.58)$$

Dosazením (2.56) a (2.57) do (2.42) dostáváme metodu

algoritmus KuM3:

$$\begin{aligned} y_k &= x_k - \frac{f(x_k)}{f[w_k, x_k]} \\ z_k &= x_k - \frac{f(x_k) + f(y_k)}{f[w_k, x_k]} \\ x_{k+1} &= z_k - \frac{f(z_k)f[w_k, x_k]}{f[x_k, y_k]f[w_k, y_k]}, \quad k = 0, 1, 2, \dots \end{aligned} \quad (2.59)$$

Odstraněním derivací došlo u této metody ke zvýšení indexu efektivity na $I_e = \sqrt[4]{5} \doteq 1,4953$, přičemž metoda zůstala 5. řádu. Přestože je hodnota indexu efektivity stále nižší, než u jiných metod, tak se osvědčila se pro řešení velmi specifických typů problémů, např. nelineární rovnice zahrnující inverzní nebo hyperbolické funkce. Více lze o této metodě dohledat v [20], včetně důkazu o řádu konvergence.

Kumarova metoda 7. řádu bez derivace

Kumar ve svém článku [19] použil úplně identický postup, jako u předcházející metody. Také na svou metodu 9. řádu (2.43). I zde všechny derivace nahradil diferencemi (2.56) a (2.57) a získal metodu ve tvaru

algoritmus KuM4:

$$\begin{aligned} y_k &= x_k - \frac{f(x_k)}{f[w_k, x_k]} \\ z_k &= y_k - \left[1 + \left(\frac{f(y_k)}{f(x_k)} \right)^2 \right] \cdot \frac{f(y_k)f[x_k, w_k]}{f[y_k, x_k]f[y_k, w_k]} \\ x_{k+1} &= z_k - \left[1 + 2 \left(\frac{f(y_k)}{f(x_k)} \right)^2 - 4 \left(\frac{f(z_k)}{f(y_k)} \right) \right] \cdot \frac{f(z_k)f[x_k, w_k]}{f[y_k, x_k]f[y_k, w_k]}, \\ k &= 0, 1, 2, \dots \end{aligned} \quad (2.60)$$

kde $f[*, \circ]$ je dáno vztahem (2.58).

Řád této metody se oproti původní sice snížil, metoda je 7. řádu, ale za zmínku stojí index efektivity $I_e = \sqrt[4]{7} \doteq 1,6266$, který je velmi vysoký. Dokonce je vyšší než u původní metody, která byla devátého řádu. Tato metoda je tedy případ, kdy nahrazením derivace diferencí došlo k zásadnímu zlepšení efektivity metody, jelikož se snížila výpočetní náročnost a zároveň odpadla nutnost počítat derivaci funkce. Pro důkaz konvergence a úplné odvození viz [19].

Hafizovy metody s váhovými funkcemi

Hafiz a spol. se zabývali problémem, jak vytvořit metody, které ve svém algoritmu nepoužívají derivaci, avšak jsou vyšších řádů. Za výchozí metodu si vzali modifikaci dvojité Newtonovy metody (2.7), ve které derivaci aproximovali dopřednou diferencí (2.54). Tuto aproximaci zde budeme dále značit pro přehlednost a jednoduchost $A(x_k) := \frac{f(x_k + f(x_k)) - f(x_k)}{f(x_k)} \approx f'(x_k)$. Tak byla získána metoda

$$\begin{aligned} \text{algoritmus HfM1:} \quad w_k &= x_k + f(x_k), \\ y_k &= x_k - \frac{f^2(x_k)}{f(w_k) - f(x_k)}, \\ x_{k+1} &= y_k - \frac{f(y_k)}{A(x_k)}, \quad k = 0, 1, 2, \dots \end{aligned} \quad (2.61)$$

Tato metoda je, stejně jako výchozí metoda (2.7), 3. řádu a má index efektivity $I_e = \sqrt[3]{3} \doteq 1,4422$. Cílem však bylo dostat metodu vyššího řádu bez dalšího zvyšování výpočetové náročnosti. K tomu Hafiz a spol. v [9] využili váhových funkcí

$$W_1 = \frac{4}{1 + (f[y_k, x_k]f[y_k, w_k])/A^2(x_k)} - 1 \quad (2.62)$$

$$W_2 = \frac{f^2[x_k, w_k]}{f[y_k, x_k]f[y_k, w_k]} \left(1 + \frac{f(y_k)f^2[x_k, w_k](f[x_k, w_k] - f[y_k, x_k])}{f(x_k)(f[y_k, x_k]f[y_k, w_k])^2} \right) \quad (2.63)$$

$$W_3 = \frac{f(x_k)f[y_k, x_k]f[y_k, w_k]f^2[x_k, w_k]}{f(x_k)f^2[y_k, x_k]f^2[y_k, w_k] - f(y_k)(f[x_k, w_k] - f[y_k, x_k])f^3[x_k, w_k]} \quad (2.64)$$

kde

$$f[* , \circ] = \frac{f(\circ) - f(*)}{\circ - *}, \quad * \neq \circ.$$

Nové metody pak dostaneme jako

$$\text{algoritmus HfM2:} \quad x_{k+1} = y_k - \frac{f(y_k)}{A(x_k)} \cdot W_1, \quad k = 0, 1, 2, \dots \quad (2.65)$$

$$\text{algoritmus HfM3:} \quad x_{k+1} = y_k - \frac{f(y_k)}{A(x_k)} \cdot W_2, \quad k = 0, 1, 2, \dots \quad (2.66)$$

$$\text{algoritmus HfM4:} \quad x_{k+1} = y_k - \frac{f(y_k)}{A(x_k)} \cdot W_3, \quad k = 0, 1, 2, \dots \quad (2.67)$$

kde y_k a w_k jsou stejné jako ve vzorci (2.61).

2.3. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Lze ukázat, že tyto metody už jsou 4. řádu, aniž by obsahovaly další výpočet funkční hodnoty oproti původní metodě, což dává vysoký index efektivity $I_e = \sqrt[3]{4} = 1,5874$. Pro důkaz konvergence metod viz [9].

Abychom mohli předchozí metody srovnávat s dalšími, uvedeme zde stručně ještě Renovu a Corderovu metodu, které jsou obě 4. řádu.

Renova metoda

Renova metoda je tvaru

$$\begin{aligned} \text{algoritmus RM:} \quad w_k &= x_k + f(x_k), \\ y_k &= x_k - \frac{f^2(x_k)}{f(w_k) - f(x_k)}, \\ x_{k+1} &= y_k - \frac{f(y_k)}{f[y_k, x_k] + f[y_k, w_k] - f[w_k, x_k] - (x_k - y_k)(w_k - y_k)}, \\ k &= 0, 1, 2, \dots \end{aligned} \tag{2.68}$$

kde $f[*, \circ]$ je dáno vztahem (2.58).

Metoda má index efektivity $I_e = \sqrt[3]{4} \doteq 1,5874$. Pro další informace ohledně odvození metody a důkazu konvergence viz [25].

Corderova metoda

Corderova metoda je dána algoritmem

$$\begin{aligned} \text{algoritmus CoM:} \quad w_k &= x_k + f(x_k), \\ y_k &= x_k - \frac{f^2(x_k)}{f(w_k) - f(x_k)}, \\ x_{k+1} &= y_k - \frac{f(y_k)}{\frac{af(y_k) - bf(w_k)}{y_k - w_k} + \frac{cf(y_k) - df(w_k)}{x_k - y_k}}, \quad k = 0, 1, 2, \dots \end{aligned} \tag{2.69}$$

kde $a, b, c, d \in \mathbb{R}$ jsou volitelné koeficienty takové, že $a = c = 1$, $b + d = 1$. My jsme v následném srovnání metod zvolili konkrétní hodnoty $b = d = 0,5$. Metoda má index efektivity $I_e = \sqrt[3]{4} \doteq 1,5874$. Bližší informace o metodě a jejím odvození lze najít v [2].

Další metody Newtonova typu bez derivace zde již uvádět nebudeme a to především proto, že většina z nich je považována spíše za modifikace Steffensenovy metody a tedy jsou v literatuře označovány jako metody Steffensenova typu.

2.4. Srovnání Newtonovy metody a metod Newtonova typu

V této podkapitole Newtonovu metodu a všechny metody Newtonova typu, které jsou v této práci uvedeny, otestujeme na několika nelineárních rovnicích.

Následující tabulka (2.1) obsahuje testovací funkce nelineárních rovnic, které pro další použití označujeme f_1, \dots, f_6 . U každé z rovnic je zde uveden i její kořen ξ . Pokud má daná rovnice více kořenů, vybrali jsme jeden z nich.

funkce	kořen
$f_1 = \operatorname{tgh} x - \operatorname{tg} x$	$\xi = 7,068582745628732$
$f_2 = x - \sin(\cos x) + 1$	$\xi = -0,166039051051030$
$f_3 = x^3 + 4x^2 - 10$	$\xi = 1,365230013414097$
$f_4 = 1 - \cos(3x) + \operatorname{tg}(2x) + \sin(4x)$	$\xi = 0,000000000000000$
$f_5 = \sin^2 x - x^2 + 1$	$\xi = 1,404491648215341$
$f_6 = xe^{x^2} - \sin^2 x + 3 \cos x + 5$	$\xi = -1,207647827130919$

Tabulka 2.1: Testovací funkce

Pro větší přehlednost uvádíme v tabulce (2.2) souhrn všech metod i s jejich zkratkami, řádem konvergence a indexem efektivity. Přičemž řazeny jsou v tabulce podle svého řádu konvergence a nikoli podle toho, jak byly za sebou uvedeny v textu.

Pro ukončení algoritmu se, u iteračních metod hledajících kořen jedné nelineární rovnice, používá některé z následujících kritérií:

$$|f(x_k)| < \varepsilon, \quad (2.70)$$

$$|x_{k+1} - x_k| < \varepsilon, \quad (2.71)$$

nebo

$$\left| \frac{x_{k+1} - x_k}{x_k} \right| < \varepsilon, \quad (2.72)$$

kde ε je požadovaná přesnost.

V této práci jsme použili zároveň dvě kritéria (2.70) a (2.71) a počítali jsme s přesností $\varepsilon = 10^{-15}$. U jednotlivých metod nás pak zajímá, kolik iterací bylo potřeba na nalezení kořene s požadovanou přesností, kolika to odpovídá vyhodnocení funkční hodnoty funkce, příp. derivace funkce. V tabulce srovnání už nebudeme uvádět, který kořen byl nalezen (má-li daná nelineární rovnice více kořenů).

2.4. SROVNÁNÍ NEWTONOVY METODY A METOD NEWTONOVA TYPU

Název metody	Zkratka	Řád	I_e
Metoda bisekce	MB	1	1
Metoda Regula falsi	MRF	1	1
Metoda sečen	MS	1,618	1,2720
Newtonova metoda	NM	2	1,4142
Noorova metoda 1	NoM1	2	1,2599
Noorova metoda 2	NoM2	2	1,2599
Steffensenova metoda	StM	2	1,4142
Weerakoon-Fernandova metoda	WFM	3	1,4422
Halleyova metoda	HaM	3	1,4422
Frontini-Sormaniova metoda	FSM	3	1,4422
Homeierova metoda	HoM	3	1,4422
Dvojitá Newtonova metoda 2	DNM2	3	1,4422
Potra-Ptáková metoda	PPM	3	1,4422
Newton-Steffensenova metoda	NSM	3	1,4422
Chunova metoda 1	CM1	3	1,4422
Hafizova metoda 1	HfM1	3	1,4422
Kouova metoda 4	KoM4	4	1,5874
Chunova metoda 2	CM2	4	1,5874
Chunova metoda 3	CM3	4	1,5874
Dvojitá Newtonova metoda 1	DNM1	4	1,4142
Jarrattova metoda	JM	4	1,5874
Hafizova metoda 2	HfM2	4	1,5874
Hafizova metoda 3	HfM3	4	1,5874
Hafizova metoda 4	HfM4	4	1,5874
Renova metoda	RM	4	1,5874
Corderova metoda	CoM	4	1,5874
Kouova metoda 1	KoM1	5	1,4953
Kouova metoda 2	KoM2	5	1,4953
Kouova metoda 3	KoM3	5	1,4953
Fangova metoda	FM	5	1,4953
Kumárova metoda 5. řádu	KuM1	5	1,3797
Kumárova metoda 5. řádu bez derivace	KuM3	5	1,4953
Kumárova metoda 7. řádu bez derivace	KuM4	7	1,6266
Džuničova metoda	DM	8	1,6818
Sharmova metoda	ShM	8	1,6818
Wangova metoda	WM	8	1,6818
Thukralova metoda 8. řádu	TM1	8	1,6818
Kumárova metoda 9. řádu	KuM2	9	1,5518
Thukralova metoda 9. řádu	TM2	9	1,7132

Tabulka 2.2: Přehled metod

Dalším kritériem, podle kterého lze metody porovnávat je numerický řád konvergence COC (z anglického Computational Order of Convergence).

Definice 2.4.1. Nechť x_{k-2}, x_{k-1}, x_k a x_{k+1} jsou čtyři po sobě jdoucí aproximace kořene ξ rovnice (2.1) blízké kořeni ξ . Pak *numerický řád konvergence* definujeme jako

$$COC \approx \frac{\ln |(x_{k+1} - x_k)/(x_k - x_{k-1})|}{\ln |(x_k - x_{k-1})/(x_{k-1} - x_{k-2})|}. \quad (2.73)$$

Z definice COC je zřejmé, že je definován pouze pokud algoritmus dané iterační metody provede alespoň 3 iterace. Obvykle očekáváme, že hodnota COC bude vycházet přibližně rovna řádu konvergence dané metody. Z tabulek srovnání vidíme, že pro naše testovací funkce byl tento předpoklad splněn pouze u metod nižších řádů. Dá se ovšem očekávat, že je hodnota COC velmi nepřesná, pokud neproběhlo větší množství iterací a nebo hodnoty x_{k-2}, x_{k-1}, x_k a x_{k+1} nejsou dostatečně blízko kořeni ξ . Proto u metod vyšších řádů (přibližně od řádu 5 výše) nepovažujeme spočtenou hodnotu COC za příliš vypovídající. COC lze také alternativně definovat jinými vztahy, např.

$$COC \approx \frac{\ln |(f(x_k)f'(x_{k-1}))/f'(x_k)f(x_{k-1})|}{\ln |(f(x_{k-1})f'(x_{k-2}))/f'(x_{k-1})f(x_{k-2})|}, \quad (2.74)$$

nebo

$$COC \approx \frac{\ln |f(x_k)/f(x_{k-1})|}{\ln |f(x_{k-1})/f(x_{k-2})|}. \quad (2.75)$$

Užitím těchto vztahů (2.74) a (2.75) však dostaneme obdobné výsledky, jako pro vztah (2.73) a tak byly použity pouze při nedostatečném počtu iterací potřebných ve vztahu (2.73).

V následujících tabulkách (2.3) a (2.4) nalezneme srovnání jednotlivých metod. Ve sloupci *it* je uvedeno, kolik iterací algoritmu dané metody proběhlo pro výpočet kořene dané funkce, ve sloupci *nf(x)* je pak uveden počet vyhodnocení funkční hodnoty odpovídající počtu iterací, a ve sloupci COC je uvedena hodnota numerického řádu konvergence pro danou metodu a funkci spočtená dle vztahu (2.73). U každé testovací funkce je rovněž uvedena zvolená počáteční aproximace.

Z tabulek srovnání jde podle očekávání vidět, že metody vyšších řádů potřebují obvykle k nalezení kořenu nelineární rovnice jen několik málo iterací. Pokud jsou tyto metody zároveň výpočtově efektivní, jedná se o solidní náhradu Newtonovy metody. Zřejmou nevýhodou metod vyšších řádů je však jejich nestabilita. Je tedy třeba dbát na vhodnost volby počáteční aproximace kořene.

Většina srovnávaných modifikací Newtonovy metody se ukázala být v porovnání s Newtonovou metodou rychlejší a především efektivnější, což odpovídá jejich indexu efektivity. Noorovy metody, které mají ze srovnávaných metod nejnížší index efektivity, se projeví navíc jako pomalé a méně iterací než Newtonova metoda dosáhly pouze při výpočtu nelineární rovnice f_6 při počáteční aproximaci $x_0 = -1, 3$.

Metody, které nepoužívají ve svém algoritmu derivaci funkce (připomeneme, že se jedná zejména o Steffensenovu metodu, pak Kumarovy metody KuM3 a KuM4, a Hafizovy metody HfM1-4) provedly výpočet za srovnatelný počet iterací, jako ostatní metody. Jelikož však nepočítaly v průběhu algoritmu derivaci funkce, lze je v jistém smyslu považovat za efektivnější.

Všechny výpočty byly provedeny v softwaru MATLAB R2016a.

2.4. SROVNÁNÍ NEWTONOVY METODY A METOD NEWTONOVA TYPU

funkce		f_1			f_2			f_3			f_4		
x_0		6,5			-1,5			0,5			-2		
metoda	řád	it	$nf(x)$	COC	it	$nf(x)$	COC	it	$nf(x)$	COC	it	$nf(x)$	COC
NM	2	6	12	2,000	11	22	1,999	7	14	2,000	5	10	1,998
NoM1	2	9	27	2,000	60	180	1,995	71	213	2,009	6	18	2,019
NoM2	2	9	27	2,000	108	324	2,007	71	213	2,000	6	18	2,019
StM	2	5	10	2,000	7	14	2,000	18	36	2,000	27	54	1,998
WFM	3	4	12	3,024	4	12	2,999	4	12	3,293	3	9	3,097
HaM	3	4	12	3,000	7	21	3,268	4	12	3,081	4	12	2,996
FSM	3	3	9	2,580	4	12	2,834	4	12	3,170	4	12	3,020
HoM	3	4	12	2,990	7	21	6,921	4	12	3,044	3	9	2,858
DNM2	3	5	15	3,001	10	30	3,446	38	114	2,988	4	12	3,181
PPM	3	5	15	2,998	10	30	2,982	38	114	2,987	4	12	3,181
NSM	3	4	12	2,986	5	15	3,063	4	12	3,207	3	9	3,518
CM1	3	4	12	2,991	18	54	2,990	4	12	3,047	4	12	3,089
HfM1	3	3	9	3,612	10	30	3,071	div	-	-	10	30	2,597
KoM4	4	3	9	3,350	8	24	3,818	5	15	3,583	3	9	4,616
CM2	4	3	9	5,253	12	36	2,090	4	12	5,003	3	9	4,153
CM3	4	4	12	4,506	div	-	-	6	18	3,913	3	9	5,991
DNM1	4	3	12	3,513	6	24	4,451	4	16	3,937	3	12	4,125
JM	4	3	9	3,926	45	135	3,291	3	9	4,515	3	9	3,895
HfM2	4	3	9	3,435	6	18	3,825	9	27	3,897	div	-	-
HfM3	4	3	9	3,537	3	9	2,685	6	18	4,187	div	-	-
HfM4	4	3	9	3,279	4	12	2,654	28	84	4,079	5	15	3,064
RM	4	3	9	3,709	6	18	4,096	6	18	3,838	4	12	4,007
CoM	4	4	12	2,982	14	42	2,000	div	-	-	8	24	2,002
KoM1	5	3	12	4,967	3	12	5,146	3	12	8,428	2	8	0,000
KoM2	5	2	8	4,932	4	16	5,336	3	12	7,030	3	12	5,068
KoM3	5	3	12	4,727	14	56	1,547	3	12	5,856	2	8	4,126
FM	5	3	12	4,627	7	28	4,592	div	-	-	3	12	5,647
KuM1	5	4	20	5,180	23	115	3,515	6	30	4,580	4	20	5,734
KuM3	5	2	8	3,786	4	16	1,408	5	20	4,698	div	-	-
KuM4	7	3	12	5,039	3	12	5,560	div	-	-	div	-	-
DM	8	3	12	4,658	7	28	2,490	3	12	5,503	3	12	5,101
ShM	8	2	8	4,156	5	20	1,733	div	-	-	2	8	3,358
WM	8	2	8	3,866	6	24	4,794	2	8	-	2	8	4,020
TM1	8	2	8	8,820	39	156	0,588	23	92	4,714	2	8	5,719
KuM2	9	2	10	5,354	div	-	-	4	20	3,819	3	15	9,259
TM2	9	6	24	1,000	div	-	-	div	-	-	3	12	6,173

Tabulka 2.3: Srovnání metod

2. NELINEÁRNÍ ROVNICE

funkce		f_5						f_6					
x_0		1			4			-1,3			-0,5		
metoda	řád	it	$nf(x)$	COC	it	$nf(x)$	COC	it	$nf(x)$	COC	it	$nf(x)$	COC
NM	2	6	12	2,000	6	12	2,000	5	10	1,996	10	20	1,995
NoM1	2	16	48	2,001	6	18	2,005	4	12	2,012	div	-	-
NoM2	2	16	48	2,000	6	18	2,005	4	12	2,012	div	-	-
StM	2	5	10	2,001	div	-	-	785	1570	2,006	1	2	-
WFM	3	4	12	3,045	4	12	2,960	3	9	2,992	11	33	3,113
HaM	3	4	12	3,010	5	15	3,030	3	9	3,065	4	12	2,943
FSM	3	4	12	3,030	4	12	2,872	3	9	3,008	5	15	3,027
HoM	3	3	9	3,605	3	9	2,254	3	9	3,023	4	12	3,026
DNM2	3	16	48	2,874	4	12	2,815	3	9	2,974	div	-	-
PPM	3	16	48	2,874	4	12	2,815	3	9	2,974	div	-	-
NSM	3	4	12	3,035	4	12	2,903	3	9	3,000	7	21	3,036
CM1	3	5	15	3,010	5	15	2,981	3	9	2,981	div	-	-
HfM1	3	3	9	2,463	div	-	-	396	1188	2,999	1	3	-
KoM4	4	4	12	3,813	4	12	3,877	3	9	3,965	div	-	-
CM2	4	3	9	4,441	3	9	2,417	3	9	3,980	68	204	3,962
CM3	4	3	9	4,15	4	12	4,591	3	9	4,002	div	-	-
DNM1	4	3	12	4,296	3	12	2,449	3	12	3,987	5	20	3,971
JM	4	3	9	4,249	4	12	3,883	2	6	3,640	4	12	4,041
HfM2	4	3	9	3,579	119	357	3,243	201	603	4,072	1	3	-
HfM3	4	3	9	3,618	5	15	3,638	div	-	-	div	-	-
HfM4	4	3	9	3,433	div	-	-	div	-	-	div	-	-
RM	4	3	9	3,613	div	-	-	3	9	3,980	div	-	-
CoM	4	4	12	2,058	38	114	2,000	786	2358	1,995	1	3	-
KoM1	5	3	12	5,618	3	12	3,180	2	8	3,899	8	32	6,270
KoM2	5	3	12	5,489	3	12	3,063	2	8	2,985	4	16	5,118
KoM3	5	3	12	5,532	3	12	3,853	2	8	3,321	3	12	3,697
FM	5	3	12	6,918	3	12	2,912	2	8	3,835	div	-	-
KuM1	5	5	25	5,273	4	20	2,483	3	15	3,895	div	-	-
KuM3	5	2	8	2,467	6	24	3,326	3	12	3,402	5	20	4,764
KuM4	7	3	12	5,967	5	20	4,129	4	16	5,658	div	-	-
DM	8	3	12	5,183	3	12	2,823	2	8	2,826	3	12	4,075
ShM	8	2	8	5,951	3	12	5,762	2	8	1,00	3	12	8,111
WM	8	2	8	4,362	3	12	6,098	2	8	1,146	3	12	7,428
TM1	8	3	12	11,66	3	12	5,356	2	8	0,602	div	-	-
KuM2	9	3	15	9,841	3	15	4,500	2	10	2,137	div	-	-
TM2	9	div	-	-	div	-	-	div	-	-	div	-	-

Tabulka 2.4: Srovnání metod

3. Soustavy nelineárních rovnic

3.1. Řešení soustav nelineárních rovnic

V této práci se budeme zabývat výhradně metodami pro řešení jedné soustavy n nelineárních rovnic o n neznámých, tedy

$$\begin{aligned}f_1(x_1, x_2, \dots, x_n) &= 0, \\f_2(x_1, x_2, \dots, x_n) &= 0, \\&\vdots \\f_n(x_1, x_2, \dots, x_n) &= 0.\end{aligned}$$

což lze také zjednodušeně zapsat jako

$$\mathbf{f}(\mathbf{x}) = \mathbf{o}, \tag{3.1}$$

kde $\mathbf{x} = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}$, $\mathbf{o} = (0, \dots, 0)^T$ a $\mathbf{f} = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_n(\mathbf{x}))^T$, $f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$.

Řešením této soustavy pak rozumíme každý číselný vektor $\boldsymbol{\xi}$ takový, že $\mathbf{f}(\boldsymbol{\xi}) = \mathbf{o}$. Dále budeme v této kapitole předpokládat, každá funkce $f_i(x)$, $i = 1, \dots, n$, je spojitá a má tolik spojitých derivací, kolik jich je v dané situaci zapotřebí.

Některé metody používané pro řešení jedné nelineární rovnice lze vhodně zobecnit i pro soustavy. Neplatí to však zejména pro metodu bisekce, která je u jedné rovnice užitečná pro nalezení vhodné počáteční aproximace. Pro soustavy neexistuje žádná taková univerzální metoda. Volba počáteční aproximace je tedy pro hledání kořenů soustav nelineárních rovnic obtížnější, než tomu je u jedné rovnice. Pro případ soustavy dvou rovnic lze v některých případech najít vhodnou počáteční aproximaci geometricky.

3.2. Newtonova metoda a metody Newtonova typu

Pro řešení soustav nelineárních rovnic je, podobně jako u jedné nelineární rovnice, nejznámější metodou Newtonova metoda. I v případě soustav je pak tato metoda výchozím bodem pro odvození dalších modifikací metody, které zde budeme také nazývat metody Newtonova typu.

3.2.1. Newtonova metoda

Pokud Newtonovu metodu pro řešení jedné nelineární rovnice zobecníme pro soustavu n nelineárních rovnic, dostáváme metodu

$$\text{Algoritmus NMS: } \mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{J}(\mathbf{x}_k)^{-1} \mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, 2, \dots \tag{3.2}$$

kde $\mathbf{J}(\mathbf{x}_k)$ je Jacobiho matice ve tvaru

$$\mathbf{J}(\mathbf{x}_k) = \begin{pmatrix} \frac{\partial f_1(\mathbf{x}_k)}{\partial x_1} & \frac{\partial f_1(\mathbf{x}_k)}{\partial x_2} & \dots & \frac{\partial f_1(\mathbf{x}_k)}{\partial x_n} \\ \frac{\partial f_2(\mathbf{x}_k)}{\partial x_1} & \frac{\partial f_2(\mathbf{x}_k)}{\partial x_2} & \dots & \frac{\partial f_2(\mathbf{x}_k)}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n(\mathbf{x}_k)}{\partial x_1} & \frac{\partial f_n(\mathbf{x}_k)}{\partial x_2} & \dots & \frac{\partial f_n(\mathbf{x}_k)}{\partial x_n} \end{pmatrix} \quad (3.3)$$

Pokud bychom však pracovali s předpisem (3.2), je zřejmé, že každý krok algoritmu by byl výpočtově náročný kvůli nutnosti spočítat vždy inverzní matici $\mathbf{J}(\mathbf{x}_k)^{-1}$. Proto je vhodné podobu algoritmu upravit následovně: Vektor \mathbf{x}_k převedeme na levou stranu rovnice, pak celou rovnici násobíme maticí $\mathbf{J}(\mathbf{x}_k)$ zleva, čímž dostaneme

$$\mathbf{J}(\mathbf{x}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k) = -\mathbf{f}(\mathbf{x}_k).$$

Dále pokud označíme $\mathbf{d}_k = (\mathbf{x}_{k+1} - \mathbf{x}_k) = (d_1^k, \dots, d_n^k)^T$, tak vidíme, že \mathbf{d}_k je řešením soustavy lineárních rovnic

$$\mathbf{J}(\mathbf{x}_k)\mathbf{d}_k = -\mathbf{f}(\mathbf{x}_k). \quad (3.4)$$

Po nalezení tohoto řešení vypočteme další aproximaci jako

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k. \quad (3.5)$$

Newtonova metoda pro soustavy má, podobně jako Newtonova metoda pro jednu rovnici, velmi výhodné vlastnosti. Patří mezi ně konvergence 2. řádu a jednoduchý zápis algoritmu.

Algoritmus Newtonovy metody musí při svém chodu v každém kroku řešit soustavu n lineárních rovnic a spočítat Jacobiho matici $\mathbf{J}(\mathbf{x})$, která sestává z n^2 parciálních derivací, což může být náročné pro velmi vysoké n . V takovém případě přichází na řadu některá z vhodných modifikací Newtonovy metody.

3.2.2. Metody Newtonova typu

Mluvíme-li o metodách Newtonova typu (též někdy nazývaných quasi-newtonovské metody), máme obvykle na mysli metody, v jejichž algoritmu se snažíme nějakým způsobem zlepšit výpočtovou náročnost původní metody. Ta spočívá především ve výpočtu Jacobiho matice a proto se zde budeme věnovat zejména metodám, které nějakým způsobem postihují právě výpočet Jacobiho matice. Byly odvozeny metody, v nichž se například derivace v Jacobiho matici nahrazují vhodnou aproximací pomocí difference, nebo se Jacobiho matice nepřepočítává v každém kroku celá znovu. Obecně můžeme zapsat algoritmus takových metod dvěma způsoby:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{B}_k^{-1}\mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, 2, \dots \quad (3.6)$$

kde \mathbf{B}_k je nějaká vhodná aproximace Jacobiho matice $\mathbf{J}(\mathbf{x}_k)$, nebo

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{M}_k\mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, 2, \dots \quad (3.7)$$

kde \mathbf{M}_k je nějaká vhodná aproximace inverzní Jacobiho matice $\mathbf{J}(\mathbf{x}_k)^{-1}$.

3.2. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Všechny metody zmíněné v této kapitole patří mezi tzv. metody sečen, což jsou takové metody, pro které platí

$$\mathbf{B}_k(\mathbf{x}_k - \mathbf{x}_{k-1}) = \mathbf{f}(\mathbf{x}_k) - \mathbf{f}(\mathbf{x}_{k-1}), \quad (3.8)$$

resp.

$$\mathbf{x}_k - \mathbf{x}_{k-1} = \mathbf{M}_k(\mathbf{f}(\mathbf{x}_k) - \mathbf{f}(\mathbf{x}_{k-1})) \quad (3.9)$$

pro všechna $k = 1, 2, \dots$

Stacionární Newtonova metoda

Jedná se o metodu, kde Jacobiho matici $\mathbf{J}(\mathbf{x})$ spočteme jen jednou na začátku algoritmu a pak ji dále nepřepočítáváme. Užijeme zápisu (3.6) a bereme $\mathbf{B}_k = \mathbf{J}(\mathbf{x}_0)$ pro $k = 0, 1, \dots$. Algoritmus metody pak vypadá následovně

$$\text{Algoritmus SNM: } \mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{J}(\mathbf{x}_0)^{-1} \mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, 2, \dots \quad (3.10)$$

Logicky taková metoda bude přesná pouze v případě, že se Jacobiho matice v průběhu algoritmu téměř nezmění, což nelze obecně předpokládat. Proto taková metoda není často použitelná.

Newtonova metoda s jemností p

Tato modifikace je příkladem metody, kde Jacobiho matici přepočítáváme jen každých p iterací v závislosti na zvolené jemnosti metody. Metoda má předpis

$$\text{Algoritmus NMJ: } \mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{B}_k^{-1} \mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, 2, \dots \quad (3.11)$$

kde $\mathbf{B}_k = \mathbf{J}(\mathbf{x}_k)$ pokud k je násobkem p , jinak $\mathbf{B}_k = \mathbf{B}_{k-p}$ a p je zvolená jemnost metody. Pro $p = 1$ se zřejmě jedná o klasickou Newtonovu metodu. My použijeme v numerickém porovnávání metod jemnost $p = 6$.

Dále lze ještě tuto metodu vylepšit dvěma způsoby. Jedním je, že jemnost p nevolíme jako konstantu, ale dynamicky ji měníme v průběhu algoritmu a to podle toho, jak rychle klesá hodnota $\|\mathbf{f}(\mathbf{x}_k)\|_2$, kde $\|\cdot\|_2$ značí klasickou Euklidovskou normu a $\|\mathbf{f}(\mathbf{x}_k)\|_2 = (\sum_{i=1}^n f_i^2(\mathbf{x}_k))^{\frac{1}{2}}$.

Potom je vhodné provést LU rozklad matice $\mathbf{J}(\mathbf{x}_\ell)$ jako

$$\mathbf{J}(\mathbf{x}_\ell) = \mathbf{L}_\ell \mathbf{U}_\ell, \quad \ell = 0, p, 2p, 3p, \dots$$

kde \mathbf{L}_ℓ je dolní trojúhelníková matice a \mathbf{U}_ℓ je horní trojúhelníková matice. Pro více informací o LU rozkladu matic nalezneme např. v [12]. LU rozklad matice je sice nákladný, ale provede se jen každých p kroků. Novou aproximaci \mathbf{x}_{k+1} pak dostaneme, pokud vyřešíme dvě soustavy lineárních rovnic

$$\mathbf{L}_\ell \mathbf{y} = -\mathbf{f}(\mathbf{x}_k), \quad \mathbf{U}_\ell \mathbf{d}_k = \mathbf{y}, \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k, \quad k = 0, 1, \dots, \quad \ell = 0, p, 2p, \dots$$

Jelikož se jedná o soustavy lineárních rovnic s trojúhelníkovou maticí, je jejich výpočet proveden nenáročným zpětným chodem.

Diskretizovaná Newtonova metoda

Obdobně jako v předchozí kapitole se nabízí úprava metody, kde parciální derivace v Jacobiho matici nahradíme aproximací pomocí diferenčního podílu:

$$\frac{\partial f_i(\mathbf{x})}{\partial x_j} \approx \delta_{ij}(\mathbf{x}, \mathbf{h}) := \frac{f_i(x_1, \dots, x_j + h_j, \dots, x_n) - f_i(\mathbf{x})}{h_j}, \quad i, j = 1, \dots, n. \quad (3.12)$$

kde $h_j \neq 0$, $j = 1, \dots, n$, jsou volené parametry a $\mathbf{h} = (h_1, \dots, h_n)^T$. Pokud volíme malé $h_j > 0$, vztah (3.12) vyjadřuje běžnou aproximaci dopřednou diferencí. Algoritmus diskretizované Newtonovy metody je vyjádřen jako

$$\begin{aligned} \text{Algoritmus NMD: } \mathbf{x}_{k+1} &= \mathbf{x}_k - \mathbf{B}_k^{-1} \mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, 2, \dots \\ \mathbf{B}_k &= \mathbf{D}(\mathbf{x}_k, \mathbf{h}_k), \end{aligned} \quad (3.13)$$

kde $\mathbf{D}(\mathbf{x}_k, \mathbf{h}_k)$ je matice s prvky $\delta_{ij}(\mathbf{x}, \mathbf{h})$, $i, j = 1, \dots, n$.

Zobecněná metoda sečen

Použijeme-li stejný algoritmus (3.13), jako v předchozí metodě, kde pouze ve vektoru \mathbf{h}_k dosadíme za j -tou složku $(h_j)_k = (x_j)_{k-1} - (x_j)_k$, kde k je index iterace, dostaneme algoritmus zobecněné metody sečen. Pro jednu rovnici jedné proměnné, tedy pro $n = 1$, se jedná o klasickou metodu sečen (2.53). Zobecněná metoda sečen má stejný řád konvergence jako původní metoda, tedy 1,618.

Zobecněná Steffensenova metoda

Stejně jako metodu sečen, tak i Steffensenovu metodu lze zobecnit pro soustavy. Provedeme to stejným způsobem, tentokrát však dosadíme $(h_j)_k = f_j(\mathbf{x}_k)$. Pro $n = 1$ se jedná o původní Steffensenovu metodu (2.55). I tato zobecněná metoda si zachovává řád konvergence roven dvěma.

R1U metody

Metody, pro které je rozdíl mezi \mathbf{B}_{k+1} a \mathbf{B}_k (resp. mezi \mathbf{M}_{k+1} a \mathbf{M}_k) dán maticí s hodnotí 1, zde nazýváme R1U metody (z anglického *Rank-one Update methods*). R1U metody jsou charakterizovány vztahy

$$\exists \mathbf{u}, \mathbf{v} \in \mathbb{R}^{n \times 1} : \mathbf{B}_{k+1} = \mathbf{B}_k + \mathbf{u} \mathbf{v}^T, \quad (3.14)$$

resp.

$$\exists \mathbf{u}, \mathbf{v} \in \mathbb{R}^{n \times 1} : \mathbf{M}_{k+1} = \mathbf{M}_k + \mathbf{u} \mathbf{v}^T, \quad (3.15)$$

kde $\mathbf{u} \mathbf{v}^T$ je matice s hodnotí 1.

Cílem těchto iteračních metod je s co nejmenší výpočtovou náročností spočítat novou aproximaci Jacobiánu \mathbf{B}_{k+1} z předchozí aproximace \mathbf{B}_k , ale rovněž efektivně spočítat inverzi aproximace Jacobiánu \mathbf{M}_{k+1} z předchozí aproximace \mathbf{M}_k . Vztah, podle kterého lze tento přepočít provést nyní odvodíme.

3.2. NEWTONOVA METODA A METODY NEWTONOVA TYPU

Mějme obecný vztah pro R1U metody:

$$\mathbf{N} = \mathbf{A} + \mathbf{u}\mathbf{v}^T.$$

Pokud platí

$$\mathbf{A}\mathbf{x} = \mathbf{u},$$

tak

$$\mathbf{N}\mathbf{x} = (\mathbf{A} + \mathbf{u}\mathbf{v}^T)\mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{u}\mathbf{v}^T\mathbf{x} = \mathbf{u} + \mathbf{u}(\mathbf{v}^T\mathbf{x}) = \mathbf{u}(1 + \mathbf{v}^T\mathbf{x}),$$

což dává

$$\mathbf{N}^{-1}\mathbf{u} = \frac{1}{1 + \mathbf{v}^T\mathbf{A}^{-1}\mathbf{u}}\mathbf{A}^{-1}\mathbf{u}.$$

Na druhou stranu pokud \mathbf{x} bude takové, že $\mathbf{v}^T\mathbf{A}^{-1}\mathbf{x} = 0$, tak

$$\mathbf{N}\mathbf{A}^{-1}\mathbf{x} = (\mathbf{A} + \mathbf{u}\mathbf{v}^T)\mathbf{A}^{-1}\mathbf{x} = \mathbf{A}\mathbf{A}^{-1}\mathbf{x} + \mathbf{u}\mathbf{v}^T\mathbf{A}^{-1}\mathbf{x} = \mathbf{x} + \mathbf{u}(\mathbf{v}^T\mathbf{A}^{-1}\mathbf{x}) = \mathbf{x},$$

z čehož dostáváme následující vztah mezi \mathbf{N}^{-1} a \mathbf{A}^{-1}

$$\mathbf{N}^{-1}\mathbf{x} = \mathbf{A}^{-1}\mathbf{x}.$$

Pokud tyto poznatky zobecníme, řešíme následující problém: máme zadanou matici \mathbf{C} a hledáme matici \mathbf{D} takovou, že platí následující 2 podmínky:

- $\mathbf{D}\mathbf{w} = \mathbf{z}$ pro dané sloupcové vektory \mathbf{w} a \mathbf{z} ,
- $\mathbf{D}\mathbf{y} = \mathbf{C}\mathbf{y}$, pro sloupcový vektor \mathbf{y} , který je ortogonální k danému sloupcovému vektoru \mathbf{g} .

Tyto podmínky jsou splněny pokud

$$\mathbf{D} = \mathbf{C} + \frac{(\mathbf{z} - \mathbf{C}\mathbf{w})\mathbf{g}^T}{\mathbf{g}^T\mathbf{w}}, \quad (3.16)$$

což nyní ověříme. Pokud $\mathbf{g}^T\mathbf{y} = 0$ a vztah (3.16) násobíme \mathbf{y} zprava, pak druhá podmínka je zřejmě splněna. Násobíme-li vztah (3.16) \mathbf{w} zprava, dostáváme $\mathbf{D}\mathbf{w} = \mathbf{C}\mathbf{w} + (\mathbf{z} - \mathbf{C}\mathbf{w}) = \mathbf{z}$, tedy i první podmínka je splněna a (3.16) je řešením problému.

Pokud zvolíme $\mathbf{C} = \mathbf{A}^{-1}$, $\mathbf{D} = \mathbf{N}^{-1}$, $\mathbf{w} = \mathbf{u}$, $\mathbf{z} = 1/(1 + \mathbf{v}^T\mathbf{A}^{-1}\mathbf{u})\mathbf{A}^{-1}\mathbf{u}$ a $\mathbf{g} = \mathbf{v}^T\mathbf{A}^{-1}$ a dosadíme do (3.16). Dostaneme vztah

$$\mathbf{N}^{-1} = \mathbf{A}^{-1} + \frac{\left[\left(\frac{1}{1 + \mathbf{v}^T\mathbf{A}^{-1}\mathbf{u}}\mathbf{A}^{-1}\mathbf{u} - \mathbf{A}^{-1}\mathbf{u} \right) \mathbf{v}^T \right] \mathbf{A}^{-1}}{\mathbf{v}^T\mathbf{A}^{-1}\mathbf{u}} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^T\mathbf{A}^{-1}}{1 + \mathbf{v}^T\mathbf{A}^{-1}\mathbf{u}}, \quad (3.17)$$

který je znám jako *Sherman-Morrisonova formule*.

Broydenovy metody

Jedny ze základních R1U metod jsou Broydenovy metody. Obvykle se nazývají Broydenova první a Broydenova druhá metoda, je však možné se setkat i s označením "dobrá" a "špatná" metoda. Toto označení plyne z toho, že Broydenova první metoda je obecně použitelnější pro více řešených problémů, zatímco druhá metoda je i podle slov autora mnohdy nefunkční. Přesto je možné, že je pro některé případy druhá metoda výhodnější než první.

3. SOUSTAVY NELINEÁRNÍCH ROVNIC

Z předchozích poznatků můžeme odvodit algoritmus metody. Počáteční hodnotu aproximace Jacobiánu zvolíme jako $\mathbf{B}_0 = \mathbf{J}(\mathbf{x}_0)$ a použijeme algoritmus Newtonovy metody pro výpočet \mathbf{x}_1 jako

$$\mathbf{B}_0 \mathbf{d}_0 = -\mathbf{f}(\mathbf{x}_0), \quad \mathbf{x}_1 = \mathbf{x}_0 + \mathbf{d}_0.$$

Pro další krok hledáme aproximaci Jacobiánu \mathbf{B}_1 , která bude splňovat podmínky rovnice (3.8), tj

$$\mathbf{B}_1(\mathbf{x}_1 - \mathbf{x}_0) = \mathbf{f}(\mathbf{x}_1) - \mathbf{f}(\mathbf{x}_0),$$

a dále pokud $\mathbf{y}^\top(\mathbf{x}_1 - \mathbf{x}_0) = 0$, pak $\mathbf{B}_1 \mathbf{y} = \mathbf{J}(\mathbf{x}_0) \mathbf{y} = \mathbf{B}_0 \mathbf{y}$.

Označíme-li $\mathbf{z}_0 = \mathbf{f}(\mathbf{x}_1) - \mathbf{f}(\mathbf{x}_0)$ a $\mathbf{d}_0 = \mathbf{x}_1 - \mathbf{x}_0$. Ze vztahu (3.16) plyne, že novou aproximaci \mathbf{B}_1 lze z \mathbf{B}_0 získat jako

$$\begin{aligned} \mathbf{B}_1 &= \mathbf{B}_0 + \frac{(\mathbf{z}_0 - \mathbf{B}_0 \mathbf{d}_0) \mathbf{d}_0^\top}{\mathbf{d}_0^\top \mathbf{d}_0} = \\ &= \mathbf{B}_0 + \frac{\mathbf{f}(\mathbf{x}_1) \mathbf{d}_0^\top}{\langle \mathbf{d}_0, \mathbf{d}_0 \rangle}. \end{aligned}$$

Tento vztah pro výpočet \mathbf{B}_1 z \mathbf{B}_0 lze snadno zobecnit pro \mathbf{B}_k , čímž dostaneme algoritmus první Broydenovy metody

$$\begin{aligned} \text{Algoritmus BGM: } \mathbf{B}_{k+1} &= \mathbf{B}_k + \frac{(\mathbf{z}_k - \mathbf{B}_k \mathbf{d}_k) \mathbf{d}_k^\top}{\mathbf{d}_k^\top \mathbf{d}_k} = \\ &= \mathbf{B}_k + \frac{\mathbf{f}(\mathbf{x}_{k+1}) \mathbf{d}_k^\top}{\langle \mathbf{d}_k, \mathbf{d}_k \rangle}, \\ \mathbf{x}_{k+1} &= \mathbf{x}_k - \mathbf{B}_k^{-1} \mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, 2, \dots \end{aligned} \tag{3.18}$$

kde $\mathbf{d}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$, $\mathbf{z}_k = \mathbf{f}(\mathbf{x}_{k+1}) - \mathbf{f}(\mathbf{x}_k)$ a \mathbf{x}_0 je počáteční aproximace. Za počáteční aproximaci Jacobiánu \mathbf{B}_0 není potřeba brát přímo Jacobián $\mathbf{J}(\mathbf{x}_0)$, lze ho např. aproximovat vhodným použitím diferencí. My jsme ale při numerických experimentech brali $\mathbf{B}_0 = \mathbf{J}(\mathbf{x}_0)$. Stejně jako u Newtonovy metody, je obvyklé nepostupovat přesně podle algoritmu (3.18), ale spočítat nejdříve \mathbf{d}_k jako řešení soustavy lineárních rovnic

$$\mathbf{B}_k \mathbf{d}_k = -\mathbf{f}(\mathbf{x}_k),$$

z čehož dostaneme další aproximaci jako

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k.$$

Může však nastat situace, kdy je naopak žádané pracovat přímo s inverzí aproximace Jacobiánu. Tím odpadá nutnost v každém kroku počítat soustavu lineárních rovnic, ale nemusí být snadné tuto inverzi získat. Pokud známe \mathbf{B}_k^{-1} , aplikací již odvozené Sherman-Morrisonovy formule (3.17) dostaneme další aproximaci jako

$$\mathbf{B}_{k+1}^{-1} = \mathbf{B}_k^{-1} - \frac{\mathbf{B}_k^{-1} \mathbf{f}(\mathbf{x}_{k+1}) \mathbf{d}_k^\top \mathbf{B}_k^{-1}}{1 + \mathbf{d}_k \mathbf{B}_k^{-1} \mathbf{f}(\mathbf{x}_{k+1})}. \tag{3.19}$$

3.3. SROVNÁNÍ METOD

Druhá Broydenova metoda je typu (3.7), což znamená, že oproti předchozím uvedeným metodám pracuje algoritmus s aproximací inverzní Jacobiho matice $\mathbf{M}_k \approx \mathbf{J}^{-1}(\mathbf{x}_k)$.

$$\begin{aligned} \text{Algoritmus BBM: } \mathbf{M}_{k+1} &= \mathbf{M}_k + \frac{(\mathbf{d}_k - \mathbf{M}_k \mathbf{y}_k) \mathbf{y}_k^T}{\langle \mathbf{y}_k, \mathbf{y}_k \rangle}, \\ \mathbf{x}_{k+1} &= \mathbf{x}_k - \mathbf{M}_k \mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, 2, \dots \end{aligned} \quad (3.20)$$

Jak už však bylo řečeno výše, metoda je značně nespolehlivá a proto se v praxi nepoužívá. Podrobné odvození Broydenových metod, některých jejich modifikací a také dalších metod Newtonova typu nalezneme např. v [8] a [21].

3.3. Srovnání metod

Podobně jako u řešení jedné nelineární rovnice, provedeme numerické srovnání některých uvedených metod i pro soustavy nelineárních rovnic. Porovnávat budeme konkrétně Newtonovu metodu, Stacionární Newtonovu metodu, Newtonovu metodu s jemností p , zobecněnou Steffensenovu metodu a obě Broydenovy metody.

Použitou metodu je znovu vhodné vždy opatřit maximálním počtem iterací. Je-li tento počet překročen, předpokládáme, že metoda diverguje a volíme jinou počáteční aproximaci. Pro zastavení výpočtového algoritmu znovu volíme vhodné kritérium, podobně jako pro jednu rovnici, např.

$$\| \mathbf{x}_{k+1} - \mathbf{x}_k \|_\infty \leq \varepsilon, \quad (3.21)$$

$$\| \mathbf{x}_{k+1} - \mathbf{x}_k \|_\infty \leq \varepsilon \| \mathbf{x}_k \|_\infty, \quad (3.22)$$

nebo

$$\| \mathbf{f}(\mathbf{x}_{k+1}) \|_\infty \leq \varepsilon, \quad (3.23)$$

kde ε je požadovaná přesnost metody a $\| \cdot \|_\infty$ značí nekonečno normu $\| \mathbf{x} \|_\infty = \max_{i=1 \dots n} |x_i|$. My použijeme pro všechny metody kritérium (3.23) a budeme pracovat s přesností $\varepsilon = 10^{-15}$.

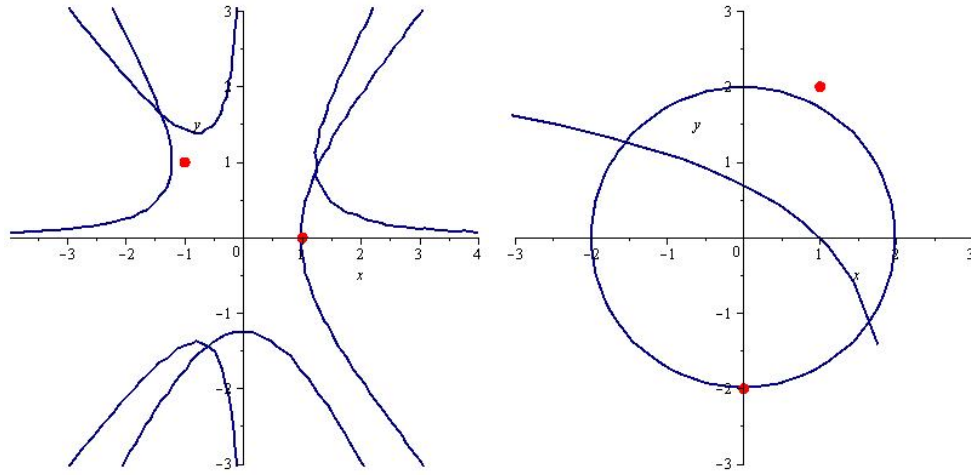
Metody otestujeme na následujících soustavách dvou rovnic o dvou neznámých:

$$\begin{aligned} \text{Soustava 1 : } f_1(x, y) &= x^3 - xy^2 - 1 = 0, \\ f_2(x, y) &= y^3 - 2x^2y + 2 = 0. \end{aligned}$$

$$\begin{aligned} \text{Soustava 2 : } f_1(x, y) &= x^2 + y^2 - 4 = 0, \\ f_2(x, y) &= \ln(2 - x) - y = 0. \end{aligned}$$

3. SOUSTAVY NELINEÁRNÍCH ROVNIC

Soustavu lze znázornit graficky, což nám pomůže při volbě počáteční aproximace, viz Obr. (3.1).



Obrázek 3.1: Vlevo soustava 1, vpravo soustava 2; tečkami vyznačeny zvolené počáteční aproximace

Jak je patrné z obrázku (3.1), tak Soustava 1 má tři kořeny $\xi_1 = [-1, 394; 1, 631]$, $\xi_2 = [1, 239; 0, 854]$ a $\xi_3 = [-0, 579; -1, 436]$ a Soustava 2 má dva kořeny $\xi_1 = [-1, 548; 1, 266]$, $\xi_2 = [1, 668; -1, 103]$. V tabulce (3.1) jsou uvedeny počty iterací, které potřebovala daná metoda pro nalezení kořene s požadovanou přesností ε .

	Soustava 1		Soustava 2	
\mathbf{x}_0	(-1,1)	(1,0)	(1,2)	(0,-2)
metoda	<i>it</i>	<i>it</i>	<i>it</i>	<i>it</i>
NMS	6	6	7	14
SNM	div	div	div	div
NMJ	17	18	36	57
ZSM	div	div	30	div
BGM	12	395	11	21
BBM	11	13	35	28

Tabulka 3.1: Srovnání metod

Z tabulky (3.1) je patrné, že Newtonova metoda konverguje vždy z uvedených metod nejrychleji. Jak již bylo ale řečeno, výhodou zde uvedených modifikací Newtonovy metody je, že oproti Newtonově metodě nepočítají v každém kroku algoritmu celý Jacobián. Proto mohou být efektivnější i přes větší množství provedených iterací.

U soustavy 1 vidíme při volbě počáteční aproximace $\mathbf{x}_0 = (1, 0)$, že Broydenova špatná metoda provedla výpočet významně rychleji než Broydenova dobrá metoda. Toto by měl být však spíše ojedinělý případ. U druhé soustavy už byla Broydenova špatná metoda podstatně pomalejší.

Zobecněná Steffensenova metoda divergovala pro většinu testovaných případů. Zdá se, že metody aproximující derivace funkce pomocí derivací jsou citlivější na volbu počáteční aproximace. U Newtonovy metody s jemností p byla zvolena jemnost $p = 6$, tedy Jacobiho matice se přepočítávala pouze jednou za 6 iterací. Stacionární Newtonova metoda se, podle očekávání, ukázala být pro zvolené testovací soustavy nepoužitelná.

3.3. *SROVNÁNÍ METOD*

Všechny výpočty byly provedeny v softwaru MATLAB R2016a.

4. Numerický výpočet integrálů

Výpočet určitého integrálu funkce $f(x)$ na intervalu $\langle a, b \rangle$ značíme jako $I(f) := \int_a^b f(x)dx$. Pokud je takový integrál neřešitelný analyticky, nebo je výpočet příliš pracný, je možné provést přibližný výpočet numericky. Takto spočtenou přibližnou hodnotu integrálu pak označíme $Q(f) := I(\varphi)$, kde $\varphi(x)$ je vhodná aproximace funkce $f(x)$.

Přibližnou hodnotu integrálu počítáme vhodnou funkcí $Q(f)$, které říkáme *kvadraturní formule*. Obecně lze kvadraturní formuli zapsat vztahem

$$Q(f) = \sum_{i=0}^n w_i f(x_i), \quad (4.1)$$

kde čísla w_i nazýváme *váhy* a body $x_i \in \langle a, b \rangle$ nazýváme *uzly* kvadraturní formule. Volbou vah a uzlů je pak určena konkrétní kvadraturní formule.

Diskretizační chyba

Nahradíme-li funkci $f(x)$ interpolačním polynomem stupně n , vzniká mezi dvěma uzly $x_{i-1}, x_i \in \langle a, b \rangle$ diskretizační chyba

$$R_i = \int_{x_{i-1}}^{x_i} (f(x) - P_n(x))dx.$$

Celková *diskretizační chyba* je potom

$$R^n(f) = I(f) - Q^n(f).$$

Algebraický řád kvadraturní formule

Kvalitu kvadraturní formule lze vyjádřit celým nezáporným číslem r , které nazýváme *algebraický řád kvadraturní formule*. Platí $R(x^j) = 0$ pro $j = 0, 1, \dots, r$ a $R(x^j) \neq 0$ pro $j > r$. To lze interpretovat tak, že kvadraturní formule řádu r integruje přesně polynomy nejvýše stupně r a polynomy vyššího stupně již přesně neintegruje.

Věta 4.0.1. *Nechť $Q_r(f) = \sum_{i=0}^{n_r} w_i^r f(x_i^r)$, $r = 0, 1, \dots$ je kvadraturní formule s kladnými koeficienty řádu r . Pak pro každou funkci $f(x)$ spojitou na intervalu $\langle a, b \rangle$ platí*

$$\lim_{r \rightarrow \infty} Q_r(f) = I(f).$$

Pro důkaz věty viz [3]. Věta nám říká, že každý integrál je možno spočítat s libovolně malou chybou, pokud zvolíme kvadraturní formuli dostatečně vysokého řádu.

4.1. Newton-Cotesovy kvadraturní formule

Jedná se o základní kvadraturní formule pro numerický výpočet integrálů. Funkci $f(x)$ aproximujeme vhodným Lagrangeovým interpolačním polynomem požadovaného stupně, který pak integrujeme. Vzdálenost mezi uzly u Newton-Cotesových kvadraturních formulí je vždy ekvidistantní, tedy pokud je dáno dělení intervalu $\langle a, b \rangle$:

$$a = x_0 < x_1 < \dots < x_{n-1} < x_n = b,$$

tak $x_{i+1} - x_i = h$, $i = 0, \dots, n-1$.

4.1. NEWTON-COTESOVY KVADRATURNÍ FORMULE

Newton-Cotesovy formule se dělí na 2 základní typy, uzavřené a otevřené. Uzavřené Newton-Cotesovy formule jsou takové, že koncové body intervalu $\langle a, b \rangle$ jsou uzly, tedy $a = x_0, b = x_n$. Pak je pro $n \geq 1, h = (b - a)/n$.

Pro otevřené Newton-Cotesovy formule platí, že koncové body intervalu naopak uzly nejsou, tedy $x_0 = a + h, x_n = b - h$ a pro $n \geq 0$ je $h = (b - a)/(n + 2)$.

Obdélníková formule

Nejjednodušší Newton-Cotesovu formuli 1. řádu dostaneme pro jediný uzel $x_0 = \frac{1}{2}(a + b)$, $n = 0$. Formule je tedy otevřená a Lagrangeův interpolační polynom $P_0(x)$ nultého stupně je

$$P_0(x) = f\left(\frac{a + b}{2}\right).$$

Z toho dostáváme kvadraturní formuli

$$Q_M(f) = \int_a^b P_0(x) dx = (b - a) f\left(\frac{a + b}{2}\right). \quad (4.2)$$

Pro přesnost je vhodné uvést, že se jedná o obdélníkovou formuli vztaženou k prostřednímu bodu intervalu, v anglické literatuře nazývanou "midpoint rule". Jinou formuli bychom dostali, pokud bychom jako uzel vybrali jeden z krajních bodů intervalu $\langle a, b \rangle$. Pak už by se ale nejednalo o otevřenou Newton-Cotesovu kvadraturní formuli, ale o polouzavřenou Newton-Cotesovu kvadraturní formuli.

Chybu obdélníkové formule lze odvodit z Taylorova rozvoje a první věty o střední hodnotě integrálu [24].

$$\begin{aligned} R_M(f) &= \int_a^b f(x) dx - \int_a^b P_0(x) dx = \\ &= \int_a^b f(x) dx - (b - a) f\left(\frac{a + b}{2}\right) = \int_a^b \left[f(x) - f\left(\frac{a + b}{2}\right) \right] dx = \\ &= \int_a^b \left[\left(x - \frac{a + b}{2}\right) f'\left(\frac{a + b}{2}\right) + \frac{1}{2} \left(x - \frac{a + b}{2}\right)^2 f''(\xi(x)) \right] dx = \\ &= \frac{1}{2} \int_a^b \left(x - \frac{a + b}{2}\right)^2 f''(\xi(x)) dx = \frac{1}{2} f''(\eta(x)) \int_a^b \left(x - \frac{a + b}{2}\right)^2 dx = \\ &= \frac{1}{24} f''(\eta)(b - a)^3, \quad \text{kde } \eta \in (a, b). \end{aligned}$$

Obecné vztahy pro chybu Newton-Cotesových kvadraturních formulí vyjadřuje následující věta

Věta 4.1.1. *Pro libovolnou Newton-Cotesovu kvadraturní formuli odpovídající sudému n lze chybu vyjádřit jako*

$$R_n(f) = \frac{M_n}{(n + 2)!} h^{n+3} f^{(n+2)}(\eta),$$

pokud $f \in C^{n+2}\langle a, b \rangle$, kde $\eta \in (a, b)$ a

$$M_n = \int_0^n t\pi_{n+1}(t) dt < 0 \quad \text{pro uzavřené formule,}$$

nebo

$$M_n = \int_{-1}^{n+1} t\pi_{n+1}(t) dt > 0 \quad \text{pro otevřené formule,}$$

kde $\pi_{n+1}(t) = \prod_{i=0}^n (t - i)$. Z toho také plyne, že algebraický řád formule je $n + 1$. Podobně pro liché n je chyba formule dána vztahem

$$R_n(f) = \frac{K_n}{(n+1)!} h^{n+2} f^{(n+1)}(\eta),$$

pokud $f \in C^{n+1}\langle a, b \rangle$, kde $\eta \in (a, b)$ a

$$K_n = \int_0^n \pi_{n+1}(t) dt < 0 \quad \text{pro uzavřené formule,}$$

nebo

$$K_n = \int_{-1}^{n+1} \pi_{n+1}(t) dt > 0 \quad \text{pro otevřené formule,}$$

Z čehož vyplývá, že algebraický řád formule je $n + 2$.

Pro důkaz věty viz [24].

Lichoběžníková formule

Lichoběžníková formule patří mezi uzavřené kvadraturní formule pro $n = 1$, kde uzly jsou krajní body intervalu $\langle a, b \rangle$: $x_0 = a, x_1 = b, h = (b - a)$. Získáme ji integrací lineárního Lagrangeova interpolačního polynomu procházejícího body $[a, f(a)]$ a $[b, f(b)]$, který je tvaru

$$\begin{aligned} P_1(x) &= \frac{x-b}{a-b}f(a) + \frac{x-a}{b-a}f(b) = \\ &= \frac{x-a-h}{-h}f(a) + \frac{x-a}{h}f(b) = \\ &= \frac{x}{h}(f(b) - f(a)) + \left(f(a) + \frac{a}{h}f(a) - \frac{a}{h}f(b)\right). \end{aligned}$$

Výsledná kvadraturní formule je potom

$$\begin{aligned} Q_T(f) &= \int_a^b P_1(x) dx = \\ &= \frac{1}{2h}(f(b) - f(a)) [x^2]_a^b + \left(f(a) + \frac{a}{h}f(a) - \frac{a}{h}f(b)\right) [x]_a^b = \\ &= \frac{1}{2h}(f(b) - f(a))(b+a)(b-a) + (b-a) \left(f(a) + \frac{a}{h}f(a) - \frac{a}{h}f(b)\right) \\ &= \frac{1}{2}(f(b) - f(a))(2a+h) + f(a)h + a(f(a) - f(b)) = \\ &= a(f(b) - f(a)) + \frac{1}{2}h(f(b) - f(a)) + hf(a) - a(f(b) - f(a)) = \\ &= \frac{1}{2}h(f(a) + f(b)). \end{aligned} \tag{4.3}$$

4.1. NEWTON-COTESOVY KVADRATURNÍ FORMULE

Chyba lichoběžníkové formule, jak vyplývá z věty (4.1.1), je

$$R_T(f) = -\frac{1}{12}f''(\eta)h^3, \quad \text{kde } \eta \in (a, b).$$

Lichoběžníková formule je shodně jako obdélníková formule 1. řádu.

Simpsonova formule

Pro $n = 2$ a 3 uzly $x_0 = a, x_1 = (a + h)$ a $x_2 = b, h = (b - a)/2$ dostaneme uzavřenou Newton-Cotesovu kvadraturní formuli, která je známá jako Simpsonova kvadraturní formule. Obdobně jako u obdélníkové a lichoběžníkové kvadraturní formule ji získáme integrací Lagrangeova interpolačního polynomu 2. stupně procházejícího body $[a, f(a)], [(a + h), f(a + h)]$ a $[b, f(b)]$, který je tvaru

$$\begin{aligned} P_2(x) &= \frac{(x - (a + h))(x - b)}{(a - (a + h))(a - b)}f(a) + \frac{(x - a)(x - b)}{((a + h) - a)((a + h) - b)}f(a + h) + \\ &+ \frac{(x - a)(x - (a + h))}{(b - a)(b - (a + h))}f(b) = \\ &= \frac{x^2 - x((a + h) + b) + (a + h)b}{h(2h)}f(a) + \frac{x^2 - x(a + b) + ab}{h(-h)}f(a + h) + \\ &+ \frac{x^2 - x(a + (a + h)) + a(a + h)}{2h(h)}f(b) = \\ &= \frac{1}{h^2} \left[x^2 \left(\frac{1}{2}f(a) - f(a + h) + \frac{1}{2}f(b) \right) + x \left(-\frac{1}{2}(2a + 3h)f(a) + (2a + 2h)f(a + h) - \right. \right. \\ &\quad \left. \left. - \frac{1}{2}(2a + h)f(b) \right) + \left(\frac{1}{2}(a + h)(a + 2h)f(a) - a(a + 2h)f(a + h) + \frac{1}{2}a(a + h)f(b) \right) \right] \end{aligned}$$

Integrací tohoto polynomu pak po zjednodušení dostáváme Simpsonovu kvadraturní formuli

$$Q_S(f) = \int_a^b P_2(x) dx = \frac{h}{3} (f(a) + 4f(a + h) + f(b)). \quad (4.4)$$

Chyba Simpsonovy formule je

$$R_S(f) = -\frac{1}{90}f^{(4)}(\eta)h^5, \quad \text{kde } \eta \in (a, b).$$

Formule je 3. řádu a je tedy významně přesnější než obdélníková nebo lichoběžníková formule.

Simpsonova 3/8 formule

Jedná se o uzavřenou Newton-Cotesovu kvadraturní formuli pro $n = 3$. Interval $\langle a, b \rangle$ je tedy rozdělen na 4 uzly se vzdáleností mezi uzly $h = (b - a)/3$. Integrací Lagrangeova interpolačního polynomu 3. stupně na intervalu $\langle a, b \rangle$ dostaneme následující kvadraturní formuli

$$Q_{S\frac{3}{8}}(f) = \frac{3}{8}h(f(a) + 3f(x_1) + 3f(x_2) + f(b)). \quad (4.5)$$

Chyba formule je dána jako

$$R_{S\frac{3}{8}}(f) = -\frac{3}{80}f^{(4)}(\eta)h^5, \quad \text{kde } \eta \in (a, b).$$

Simpsonova 3/8 formule je 3. řádu stejně jako Simpsonova formule.

Booleova formule

Booleovu formuli dostaneme pro $n = 4$, tedy pro 5 uzlů na intervalu $\langle a, b \rangle$ a vzdáleností mezi uzly $h = (b - a)/4$. Integrací příslušného interpolačního polynomu 4. stupně dostáváme formuli

$$Q_B(f) = \frac{2}{45}h(7f(a) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(b)). \quad (4.6)$$

Chyba formule je

$$R_B(f) = -\frac{8}{945}f^{(6)}(\eta)h^7, \quad \text{kde } \eta \in (a, b).$$

Booleova formule je 5. řádu.

Integrací interpolačních polynomů vyšších stupňů můžeme dostat další kvadratické formule s více uzly a vyšších řádů. Jejich váhové koeficienty pro uzavřené Newton-Cotesovy kvadraturní formule jsou v tabulce (4.1) a váhové koeficienty pro otevřené formule jsou v tabulce (4.2).

n	w_0	w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8
1	$\frac{1}{2}$	$\frac{1}{2}$	0	0	0	0	0	0	0
2	$\frac{1}{3}$	$\frac{4}{3}$	$\frac{1}{3}$	0	0	0	0	0	0
3	$\frac{3}{8}$	$\frac{9}{8}$	$\frac{9}{8}$	$\frac{3}{8}$	0	0	0	0	0
4	$\frac{14}{45}$	$\frac{64}{45}$	$\frac{24}{45}$	$\frac{64}{45}$	$\frac{14}{45}$	0	0	0	0
5	$\frac{95}{288}$	$\frac{375}{288}$	$\frac{250}{288}$	$\frac{250}{288}$	$\frac{375}{288}$	$\frac{95}{288}$	0	0	0
6	$\frac{41}{140}$	$\frac{216}{140}$	$\frac{27}{140}$	$\frac{272}{140}$	$\frac{27}{140}$	$\frac{216}{140}$	$\frac{41}{140}$	0	0
7	$\frac{5257}{17280}$	$\frac{25039}{17280}$	$\frac{9261}{17280}$	$\frac{20923}{17280}$	$\frac{20923}{17280}$	$\frac{9261}{17280}$	$\frac{25039}{17280}$	$\frac{5257}{17280}$	0
8	$\frac{3956}{14175}$	$\frac{23552}{14175}$	$-\frac{3712}{14175}$	$\frac{41984}{14175}$	$-\frac{18160}{14175}$	$\frac{41984}{14175}$	$-\frac{3712}{14175}$	$\frac{23552}{14175}$	$\frac{3956}{14175}$

Tabulka 4.1: Váhové koeficienty uzavřených Newton-Cotesových kvadraturních formulí

Další Newton-Cotesovy kvadraturní formule vyšších řádů lze nalézt např v [32]. V tabulkách (4.1) a (4.2) ale vidíme, že váhové koeficienty některých Newton-Cotesových kvadraturních formulí mohou nabývat záporných hodnot. To platí zejména pro otevřené formule pro $n \geq 2$, ale také pro uzavřené formule pro $n \geq 8$. Tento fakt může mít zásadní vliv na stabilitu těchto formulí, neboť dochází při výpočtu k zaokrouhlovacím chybám. Proto se Newton-Cotesovy kvadraturní formule se zápornými váhovými koeficienty nepoužívají. Místo nich lze užít Gaussovy kvadraturní formule, které nemají ekvidistantní vzdálenost mezi uzly. Gaussovým kvadraturním formulím se však v této práci věnovat nebudeme.

4.2. SLOŽENÉ NEWTON-COTESOVY KVADRATURNÍ FORMULE

n	w_0	w_1	w_2	w_3	w_4	w_5	w_6
1	2	0	0	0	0	0	0
2	$\frac{3}{2}$	$\frac{3}{2}$	0	0	0	0	0
3	$\frac{8}{3}$	$-\frac{4}{3}$	$\frac{8}{3}$	0	0	0	0
4	$\frac{55}{24}$	$\frac{5}{24}$	$\frac{5}{24}$	$\frac{55}{24}$	0	0	0
5	$\frac{66}{20}$	$-\frac{84}{20}$	$\frac{156}{20}$	$-\frac{84}{20}$	$\frac{66}{20}$	0	0
6	$\frac{4277}{1440}$	$-\frac{3171}{1440}$	$\frac{3934}{1440}$	$\frac{3934}{1440}$	$-\frac{3171}{1440}$	$\frac{4277}{1440}$	0
7	$\frac{3680}{945}$	$-\frac{7632}{945}$	$\frac{17568}{945}$	$-\frac{19672}{945}$	$\frac{17568}{945}$	$-\frac{7632}{945}$	$\frac{3680}{945}$

Tabulka 4.2: Váhové koeficienty otevřených Newton-Cotesových kvadraturních formulí

4.2. Složené Newton-Cotesovy kvadraturní formule

Protože jsou Newton-Cotesovy kvadraturní formule vyšších řádů nepoužitelné, tak pro zpřesnění výpočtu je obvyklé interval $\langle a, b \rangle$ rozdělit na m podintervalů $\langle a_i, b_i \rangle$, $i = 1, 2, \dots, m$ tak, že $a = a_1 < b_1 = a_2 < \dots < b_{m-1} = a_m < b_m = b$ a na každý z těchto podintervalů použít kvadraturní formuli $Q_i(f)$, obvykle stejného typu. Složená kvadraturní formule je pak dána jako součet těchto dílčích kvadraturních formulí, tedy $Q^m(f) = \sum_{i=1}^m Q_i(f)$. Platí, že pokud délku podintervalů zmenšujeme, výsledná kvadraturní formule $Q^m(f)$ se přibližuje přesné hodnotě integrálu $I(f)$, o čemž pojednává následující věta.

Věta 4.2.1. *Nechť $a = a_1 < b_1 = a_2 < \dots < b_{m-1} = a_m < b_m = b$ je dělení intervalu $\langle a, b \rangle$, $\sigma = \max_i(b_i - a_i)$, $Q^m(f) = \sum_{i=1}^m Q_i(f)$ je složená kvadraturní formule na $\langle a, b \rangle$ a $Q_i(f) = \sum_{j=0}^{n_i} w_j^i f(x_j^i)$ je kvadraturní formule na $\langle a_i, b_i \rangle$, která má kladné koeficienty a je řádu alespoň 0. Pak pro každou funkci $f(x)$, která je na intervalu $\langle a, b \rangle$ lipschitzovsky spojitá, platí*

$$\lim_{\sigma \rightarrow 0} Q^m(f) = I(f).$$

Důkaz věty nalezneme v [3].

Složená Simpsonova formule

Výše uvedeným postupem nyní ukážeme odvození složené Simpsonovy formule. Interval $\langle a, b \rangle$ rozdělíme na m dílků. Vzdálenost 2 uzlů je pak $h = \frac{(b-a)}{m}$. Dále sčítáme Simpsonovy formule na intervalech $\langle x_0, x_1 \rangle, \langle x_1, x_2 \rangle, \dots, \langle x_{m-1}, x_m \rangle$. Pro přehlednost budeme značit $f_i = f(x_i)$ a $f_{i+\frac{1}{2}} = f(x_i + \frac{h}{2})$. Pak dostaneme vztah

$$\begin{aligned} Q_S^m(f) &= \frac{h}{6}(f_0 + 4f_{\frac{1}{2}} + f_1) + \frac{h}{6}(f_1 + 4f_{\frac{3}{2}} + f_2) + \dots + \frac{h}{6}(f_{m-1} + 4f_{m-\frac{1}{2}} + f_m) = \\ &= \frac{1}{6}h \left(f_0 + 4f_{\frac{1}{2}} + 2f_1 + 4f_{\frac{3}{2}} + 2f_2 + \dots + 2f_{m-1} + 4f_{m-\frac{1}{2}} + f_m \right). \end{aligned} \quad (4.7)$$

Chybu této formule dostaneme jako

$$\begin{aligned}
 R_S^m(f) &= -\frac{1}{90} \sum_{i=0}^{m-1} \left(\frac{h}{2}\right)^5 f^{(4)}(\eta_i) = \\
 &= -\frac{1}{90} \frac{h^5}{2^5} m \left(\frac{1}{m} \sum_{i=0}^{m-1} f^{(4)}(\eta_i) \right) = \\
 &= -\frac{1}{90} \frac{b-a}{32} h^4 f^{(4)}(\eta) = \\
 &= -\frac{b-a}{2880} h^4 f^{(4)}(\eta), \quad \text{kde } \eta \in (a, b).
 \end{aligned}$$

O celkové diskretizační chybě složených Newton-Cotesových kvadraturních formulí pojednává následující věta.

Věta 4.2.2. *Nechť je použita složená Newton-Cotesova kvadraturní formule pro n sudé. Pokud $f \in C^{n+2}\langle a, b \rangle$, tak pro diskretizační chybu formule platí*

$$R_m^n(f) = \frac{(b-a)M_n}{(n+2)^{n+3}(n+2)!} h^{n+2} f^{(n+2)}(\eta),$$

pro složené otevřené formule a

$$R_m^n(f) = \frac{(b-a)M_n}{n^{n+3}(n+2)!} h^{n+2} f^{(n+2)}(\eta),$$

pro složené uzavřené formule, kde $\eta \in \langle a, b \rangle$ a M_n je stejné jako ve větě (4.2.2). Z toho plyne, že algebraický řád formule je $n+2$.

Obdobně pro složené Newton-Cotesovy kvadraturní formule pro n liché platí: pokud $f \in C^{n+1}\langle a, b \rangle$, tak

$$R_m^n(f) = \frac{(b-a)K_n}{(n+2)^{n+2}(n+1)!} h^{n+1} f^{(n+1)}(\eta),$$

pro složené otevřené formule a

$$R_m^n(f) = \frac{(b-a)K_n}{n^{n+2}(n+1)!} h^{n+1} f^{(n+1)}(\eta),$$

pro složené uzavřené formule, kde $\eta \in \langle a, b \rangle$ a K_n je stejné jako ve větě (4.2.2). Z toho plyne, že řád formule je $n+1$.

Pro důkaz věty viz [24].

Stručně uvedeme ještě další základní složené Newton-Cotesovy kvadraturní formule a příslušné vztahy pro výpočet diskretizační chyby. Postup při odvozování je analogický jako u složené Simpsonovy kvadraturní formule.

4.3. SROVNÁNÍ NEWTON-COTESOVÝCH KVADRATURNÍCH FORMULÍ

Složená obdélníková formule

$$Q_M^m(f) = h \sum_{i=1}^m f_{i-\frac{1}{2}h} = f_{\frac{1}{2}} + f_{\frac{3}{2}} + \dots + f_{m-\frac{3}{2}} + f_{m-\frac{1}{2}}, \quad (4.8)$$

$$\text{kde } h = \frac{b-a}{m},$$

$$R_M^m = \frac{b-a}{24} f''(\eta) h^2, \quad \text{kde } \eta \in (a, b).$$

Složená lichoběžníková formule

$$Q_T^m(f) = \sum_{i=1}^m \frac{1}{2} h (f_{i-1} + f_i) = \frac{1}{2} f_0 + f_1 + \dots + f_{m-1} + \frac{1}{2} f_m, \quad (4.9)$$

$$\text{kde } h = \frac{b-a}{m},$$

$$R_T^m = -\frac{b-a}{12} f''(\eta) h^2, \quad \text{kde } \eta \in (a, b).$$

Složená Booleova formule

$$\begin{aligned} Q_B^m(f) &= \frac{h}{90} \sum_{i=0}^{m-1} \left(7f_i + 32f_{i+\frac{1}{4}} + 12f_{i+\frac{1}{2}} + 32f_{i+\frac{3}{4}} + 7f_{i+1} \right) = \\ &= \frac{h}{90} \left[7(f_0 + f_m) + \sum_{i=0}^m \left[32(f_{i+\frac{1}{4}} + f_{i+\frac{3}{4}}) + 12f_{i+\frac{1}{2}} \right] + \sum_{i=1}^{m-1} 14f_i \right], \end{aligned} \quad (4.10)$$

$$\text{kde } h = \frac{b-a}{m},$$

$$R_B^m = -\frac{b-a}{1935360} f^{(6)}(\eta) h^6, \quad \text{kde } \eta \in (a, b).$$

Obecně se složené Newton-Cotesovy kvadraturní formule pro vyšší n nepoužívají, protože jsou, stejně jako jednoduché Newton-Cotesovy kvadraturní formule, často nestabilní z důvodu zaokrouhlovacích chyb.

4.3. Srovnání Newton-Cotesových kvadraturních formulí

Pro demonstraci rozdílnosti jednotlivých kvadraturních formulí nyní některé vybrané formule porovnáme aplikací na několik testovacích funkcí. Použijeme uzavřené Newton-Cotesovy kvadraturní formule pro $n = 1, \dots, 6$ a pak také základní složené Newton-Cotesovy kvadraturní formule - obdélníkovou, lichoběžníkovou, Simpsonovu a Booleovu. Složené formule testujeme nejprve pro dělení na 5 intervalů a poté na 15 intervalů. Otevřené Newton-Cotesovy formule se prakticky nepoužívají a proto je ani v tomto srovnávání nebudeme používat.

4. NUMERICKÝ VÝPOČET INTEGRÁLŮ

Jako testovací funkce bereme

$$f_1 = \frac{6}{3+x^2} \quad \text{na intervalu } \langle -5, 5 \rangle,$$

$$f_2 = x^{\frac{7}{3}} \quad \text{na intervalu } \langle 0, 3 \rangle.$$

V tabulkách vždy uvedeme nejprve přesnou hodnotu integrálu dané funkce na daném intervalu, pak hodnoty spočítané jednotlivými kvadraturními formullemi. Ve vedlejším sloupci rovněž uvedeme chybu dané formule.

funkce	f_1		f_2	
interval	$\langle -5, 5 \rangle$		$\langle 0, 3 \rangle$	
$I(f)$	8,57242628		11,68222152	
n	$Q(f)$	$R(f)$	$Q(f)$	$R(f)$
1	2,14285714	6,42956914	19,47036920	-7,78814768
2	14,04761905	-5,47519277	11,64133716	0,04088436
3	8,32417582	0,24825046	11,66223702	0,01998450
4	7,61261261	0,95981367	11,68010486	0,00211666
5	8,09523810	0,47718818	11,68085614	0,00136538
6	9,53968326	-0,96725698	11,68185143	0,00037009

Tabulka 4.3: Srovnání uzavřených Newton-Cotesových kvadraturních formulí

funkce	f_1		f_2	
interval	$\langle -5, 5 \rangle$		$\langle 0, 3 \rangle$	
$I(f)$	8,57242628		11,68222152	
	$Q(f)$	$R(f)$	$Q(f)$	$R(f)$
obdélníková formule	8,69172932	-0,11930304	11,52984679	0,15237473
lichoběžníková formule	8,42857000	0,14385628	11,98630000	-0,30407848
Simpsonova formule	8,60401003	-0,03158375	11,68198885	0,00023267
Booleova formule	8,57015666	0,00226962	11,68221153	0,00000999

Tabulka 4.4: Srovnání složených Newton-Cotesových kvadraturních formulí pro 5 intervalů

funkce	f_1		f_2	
interval	$\langle -5, 5 \rangle$		$\langle 0, 3 \rangle$	
$I(f)$	8,57242628		11,68222152	
	$Q(f)$	$R(f)$	$Q(f)$	$R(f)$
obdélníková formule	8,57525024	-0,00282396	11,66537013	0,01685139
lichoběžníková formule	8,56677000	0,00565628	11,71590000	-0,03367848
Simpsonova formule	8,57242336	0,00000292	11,68221525	0,00000627
Booleova formule	8,57242624	0,00000004	11,68222126	0,00000026

Tabulka 4.5: Srovnání složených Newton-Cotesových kvadraturních formulí pro 15 intervalů

V tabulce (4.3) je patrné, že jednoduché Newton-Cotesovy kvadraturní formule pro malý počet uzlů nejsou pro výpočet integrálů vhodné, pokud je použijeme na delším

4.3. SROVNÁNÍ NEWTON-COTESOVÝCH KVADRATURNÍCH FORMULÍ

intervalu. Funkce f_1 se ukázala být problematická i pro formule s větším počtem uzlů, neboť i formule pro $n = 6$ se dopouští relativně velké chyby.

Oproti tomu složené kvadraturní formule se ukázaly být v našem případě ideální volbou pro výpočet daných integrálů. Pro malé dělení intervalu se dopouští významné chyby pouze obdélníková a lichoběžníková formule. Pokud ale zvětšujeme množství intervalů, na které dělíme původní interval, je možné použít i složenou obdélníkovou kvadraturní formuli s dostatečnou přesností.

Všechny výpočty byly provedeny v softwaru MATLAB R2016a.

5. Závěr

Cílem práce bylo uvedení Newtonových numerických metod, jejich aplikace na příkladech a porovnání s dalšími adekvátními metodami. Nejprve byla obecně popsána problematika řešení nelineárních rovnic a uvedena Newtonova metoda, která je pro tento problém asi nejznámější numerickou metodou. Dále jsme zde uvedli větší množství metod Newtonova typu, tedy těch, které vychází z původní Newtonovy metody. Tyto metody byly rozděleny do dvou hlavních skupin: na metody s derivací a metody bez derivace. Bylo vysvětleno, že mohou být metody bez derivace významně efektivnější v případech, kdy je výpočet derivace příliš náročný. Metody byly následně porovnány na několika testovacích funkcích. Mnoho metod vyšlo ve srovnání s Newtonovou metodou o něco lépe, jak co se týče rychlosti konvergence metod, tak jejich výpočtové efektivity. Přesto se Newtonova metoda ukázala být pro většinu řešených funkcí dostatečná. Její výhoda však spočívá především v jednoduchosti a spolehlivosti oproti dalším metodám.

V další kapitole byl problém řešení nelineárních rovnic zobecněn na soustavy n nelineárních rovnic o n neznámých. Bylo ukázáno, že Newtonovu metodu lze zobecnit na soustavy a dále byly ukázány některé její modifikace. V případě soustav byla Newtonova metoda ze všech porovnávaných metod nejrychleji konvergující a hlavně její použití vždy vedlo k nalezení řešení. Ostatní metody byly však uvedeny hlavně proto, že mohou být významně výpočtově úspornější, pokud je náročné spočítat Jacobiho matici pro danou soustavu.

Poslední kapitola byla věnována problematice numerického výpočtu integrálů. Byly zde odvozeny Newton-Cotesovy kvadrurní formule, i složené Newton-Cotesovy kvadrurní formule, které pak byly v závěru kapitoly aplikovány na několika příkladech. Zde se ukázalo, že je výhodnější používat složené kvadrurní formule aplikované na vhodně děleném intervalu.

Na práci je možno dále navázat zpracováním dalších Newtonových metod, např. v oblasti optimalizace, interpolace aj. a opět u nich uvést i jejich modifikace. Metody Newtonova typu jsou v současnosti stále rozvíjeny a zdokonalovány, jelikož možnost aplikace takových metod je velmi široká v celé řadě vědních oborů.

Literatura

- [1] ARGYROS, I. K., Dong CHEN a Qingshan QIAN. The Jarratt method in Banach space setting. *Journal of Computational and Applied Mathematics* [online]. 1994, (51), 103-106 [cit. 2017-05-20]. Dostupné z: [https://doi.org/10.1016/0377-0427\(94\)90093-0](https://doi.org/10.1016/0377-0427(94)90093-0)
- [2] CORDERO, Alicia a Juan R. TORREGROSA. A class of Steffensen type methods with optimal order of convergence. *Applied Mathematics and Computation* [online]. 2011, (217), 7653-7659 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.amc.2011.02.067>
- [3] ČERMÁK, Libor a Rudolf HLAVIČKA. *Numerické metody*. Vyd. 2. Brno: Akademické nakladatelství CERM, 2008. ISBN 978-80-214-3752-4.
- [4] DŽUNIĆ, Jovana a Miodrag S. PETKOVIĆ. A Family of Three-Point Methods of Ostrowski's Type for Solving Nonlinear Equations. *Journal of Applied Mathematics* [online]. 2011 [cit. 2017-05-20]. Dostupné z: <http://dx.doi.org/10.1155/2012/425867>
- [5] DŽUNIĆ, Jovana a Miodrag S. PETKOVIĆ. A Cubically Convergent Steffensen-Like Method for Solving Nonlinear Equations. *Applied Mathematics Letters* [online]. 2012, **25**(11), 1881-1886 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.aml.2012.02.055>
- [6] FANG, Liang, Li SUN a Guoping HE. An efficient Newton-type method with fifth-order convergence for solving nonlinear equations. *Computational & Applied Mathematics* [online]. 2008, **27**(3) [cit. 2017-05-20]. Dostupné z: <http://ref.scielo.org/hqxx45>
- [7] FRONTINI, M. a E. SORMANI. Some variant of Newton's method with third-order convergence. *Applied Mathematics and Computation* [online]. 2003, **140**(2-3), 419-426 [cit. 2017-05-20]. Dostupné z: [https://doi.org/10.1016/S0096-3003\(02\)00238-2](https://doi.org/10.1016/S0096-3003(02)00238-2)
- [8] HAELTERMAN, Robby. *Analytical study of the Least Squares Quasi-Newton method for interaction problems*. Gent, 2008. Disertace. Univerzita Gent. Vedoucí práce Prof. Dr. Ir. Jan Vierendeels.
- [9] HAFIZ, M.A. Solving Nonlinear Equations Using Steffensen-Type Methods With Optimal Order of Convergence. *Palestine Journal of Mathematics* [online]. 2014, **3**(1), 114-119 [cit. 2017-05-20]. Dostupné z: <https://pdfs.semanticscholar.org/601d/>
- [10] HANSON, Floyd. Convergence of Secant Method. In: *MCS 471 Class Notes* [online]. University of Illinois at Chicago, 2005 [cit. 2017-05-20]. Dostupné z: <http://www.math.uic.edu/~hanson/mcs471/classnotes.html>
- [11] HOMEIER, H.H.H. On Newton-type methods with cubic convergence. *Journal of Computational and Applied Mathematics* [online]. 2005, **176**(2), 425-432 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.cam.2004.07.027>
- [12] HOROVÁ, Ivana a Jiří ZELINKA. *Numerické metody*. 2., rozš. vyd. Brno: Masarykova univerzita v Brně, 2004. ISBN 80-210-3317-7.

- [13] HOŠKOVÁ, Šárka, Jaromír KUBEN a Pavlína RAČKOVÁ. *Integrální počet funkcí jedné proměnné* [online]. Ostrava, 2006 [cit. 2017-04-20]. Dostupné z: <http://www.math.muni.cz/~plch/ip.pdf>
- [14] CHUN, Changbum. A family of composite fourth-order iterative methods for solving nonlinear equations. *Applied Mathematics and Computation* [online]. 2007, **187**(2), 951–956 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.amc.2006.09.009>
- [15] CHUN, Changbum. *Construction of third-order iterative families free from second derivatives for solving nonlinear equations*. Korea University of Technology and Education, Cheonan City.
- [16] KINCAID, David a E. W. CHENEY. *Numerical analysis: mathematics of scientific computing*. 3rd ed. Providence, R.I.: American Mathematical Society, 2009. Sally series (Providence, R.I.), 2. ISBN 978-0-8218-4788-6.
- [17] KOU, Jisheng, Yitian LI a Xiuhua WANG. Some modifications of Newton's method with fifth-order convergence. *Journal of Computational and Applied Mathematics* [online]. 2007, **209**(2), 146–152 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.cam.2006.10.072>
- [18] KOU, Jisheng, Yitian LI a Xiuhua WANG. A composite fourth-order iterative method for solving non-linear equations. *Applied Mathematics and Computation* [online]. 2007, **184**(2), 471–475 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.amc.2006.05.181>
- [19] KUMAR, Manoj, Akhilesh Kumar SINGH a Akanksha SRIVASTAVA. Various Newton-type iterative methods for solving nonlinear equations. *Journal of the Egyptian Mathematical Society* [online]. 2013, **21**(3), 334–339 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.joems.2013.03.001>
- [20] KUMAR, Manoj, Akhilesh Kumar SINGH a Akanksha SRIVASTAVA. A New Fifth Order Derivative Free Newton-Type Method for Solving Nonlinear Equations. *Applied Mathematics & Information Sciences* [online]. 2015, **9**(3), 1507–1513 [cit. 2017-05-20]. Dostupné z: <http://www.naturalspublishing.com/files/published/ahb21733nf19a5.pdf>
- [21] MARTÍNEZ, José Mario. Practical quasi-Newton methods for solving nonlinear systems. *Journal of Computational and Applied Mathematics* [online]. 2000, **124**(1-2), 97–121 [cit. 2017-05-20]. Dostupné z: [https://doi.org/10.1016/S0377-0427\(00\)00434-9](https://doi.org/10.1016/S0377-0427(00)00434-9)
- [22] NOOR, Muhammad Aslam, Khalida Inayat NOOR a Kshif AFTAB. Some New Iterative Methods for Solving Nonlinear Equations. *World Applied Sciences Journal* [online]. 2012, **20**(6), 870–874 [cit. 2017-05-20]. ISSN 1818-4952.
- [23] POTRA, F.-A. a V. PTÁK. *Nondiscrete induction and iterative processes*. Boston: Pitman Advanced Pub. Program, 1984. ISBN 02-730-8627-8.
- [24] QUARTERONI, Alfio, Riccardo SACCO a Fausto SALERI. *Numerical Mathematics*. 2nd ed. New York, NY: Springer, 2006. ISBN 978-038-7227-504.

LITERATURA

- [25] REN, Hongmin, Qingbiao WU a Weihong BI. A class of two-step Steffensen type methods with fourth-order convergence. *Applied Mathematics and Computation* [online]. 2009, **209**(2), 206–210 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.amc.2008.12.039>
- [26] SHARMA, Janak Raj. A composite third order Newton–Steffensen method for solving nonlinear equations. *Applied Mathematics and Computation* [online]. 2005, **169**(1), 242–246 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.amc.2004.10.040>
- [27] SHARMA, Janak Raj a Rajni SHARMA. A new family of modified Ostrowski’s methods with accelerated eighth order convergence. *Numerical Algorithms* [online]. 2010, **54**(4), 445–458 [cit. 2017-05-20]. DOI: 10.1007/s11075-009-9345-5. ISSN 1017-1398. Dostupné z: <http://link.springer.com/10.1007/s11075-009-9345-5>
- [28] THUKRAL, R. A Newton-Type Method With Ninth-Order Convergence for Solving Nonlinear Equations. *Journal of Numerical Mathematics and Stochastics* [online]. 2016, **8**(1), 09–16 [cit. 2017-05-20]. ISSN 2151-2302. Dostupné z: <http://www.jnmas.org/jnmas8-2.pdf>
- [29] THUKRAL, R. New modifications of Newton-type methods with eighth-order convergence for solving nonlinear equations. *Journal of Advances in Mathematics*. 2015, **10**(3), 3362–3373.
- [30] WANG, Xia a Liping LIU. Modified Ostrowski’s method with eighth-order convergence and high efficiency index. *Applied Mathematics Letters* [online]. 2010, **23**(5), 549–554 [cit. 2017-05-20]. Dostupné z: <https://doi.org/10.1016/j.aml.2010.01.009>
- [31] WEERAKOON, S. a T.G.I. FERNANDO. A variant of Newton’s method with accelerated third-order convergence. *Applied Mathematics Letters* [online]. 2000, **13**(8), 87–93 [cit. 2017-05-20]. Dostupné z: [https://doi.org/10.1016/S0893-9659\(00\)00100-2](https://doi.org/10.1016/S0893-9659(00)00100-2)
- [32] WEISSTEIN, Eric W. Newton-Cotes Formulas. *MathWorld–A Wolfram Web Resource* [online]. [cit. 2017-05-20]. Dostupné z: <http://mathworld.wolfram.com/Newton-CotesFormulas.html>

6. Seznam použitých zkratek a symbolů

ξ	kořen nelineární rovnice
ε	výpočtová přesnost
$\langle a, b \rangle$	uzavřený interval
$C\langle a, b \rangle$	množina funkcí spojitých na intervalu $\langle a, b \rangle$
$C^n\langle a, b \rangle$	množina funkcí se spojitými derivacemi do řádu n včetně na intervalu $\langle a, b \rangle$.
I_e	index efektivity metody
COC	numerický řád konvergence metody
\mathbf{a}	vektor
\mathbf{A}	matice
$\mathbf{J}(\mathbf{x})$	Jacobiho matice
$\ \cdot\ _2$	Euklidovská norma vektoru
$\ \cdot\ _\infty$	nekonečno norma vektoru
$\langle \mathbf{a}, \mathbf{b} \rangle$	skalární součin vektorů \mathbf{a} a \mathbf{b}
$Q(f)$	kvadrurní formule
$R(f)$	chyba kvadrurní formule
$P_n(x)$	Lagrangeův interpolační polynom n -tého stupně